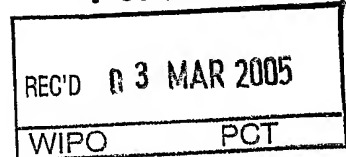


PCT/JP 2005/002527

10.02.2005

日 本 国 特 許 庁  
JAPAN PATENT OFFICE



別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出 願 年 月 日                      2 0 0 4 年    3 月 2 2 日  
Date of Application:

出 願 番 号                      特 願 2 0 0 4 - 0 8 2 7 0 9  
Application Number:  
[ST. 10/C] :                      [ J P 2 0 0 4 - 0 8 2 7 0 9 ]

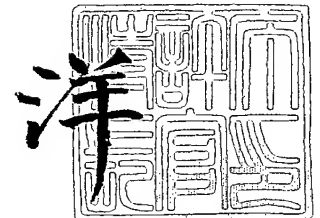
出      願      人                      セイコーエプソン株式会社  
Applicant(s):

**PRIORITY DOCUMENT**  
SUBMITTED OR TRANSMITTED IN  
COMPLIANCE WITH  
RULE 17.1(a) OR (b)

2 0 0 4 年 1 2 月    8 日

特許庁長官  
Commissioner,  
Japan Patent Office

小 川



出証番号    出証特 2 0 0 4 - 3 1 1 2 0 5 7

【書類名】 特許願  
【整理番号】 J0107361  
【あて先】 特許庁長官 殿  
【国際特許分類】 H04N 1/60  
【発明者】  
    【住所又は居所】 長野県諏訪市大和三丁目 3 番 5 号 セイコーエプソン株式会社内  
    【氏名】 角谷 繁明  
【特許出願人】  
    【識別番号】 000002369  
    【氏名又は名称】 セイコーエプソン株式会社  
【代理人】  
    【識別番号】 100095728  
    【弁理士】  
    【氏名又は名称】 上柳 雅誉  
    【連絡先】 0 2 6 6 - 5 2 - 3 5 2 8  
【選任した代理人】  
    【識別番号】 100107076  
    【弁理士】  
    【氏名又は名称】 藤網 英吉  
【選任した代理人】  
    【識別番号】 100107261  
    【弁理士】  
    【氏名又は名称】 須澤 修  
【手数料の表示】  
    【予納台帳番号】 013044  
    【納付金額】 21,000円  
【提出物件の目録】  
    【物件名】 特許請求の範囲 1  
    【物件名】 明細書 1  
    【物件名】 図面 1  
    【物件名】 要約書 1  
    【包括委任状番号】 0109826

**【書類名】 特許請求の範囲****【請求項 1】**

画像データに所定の画像処理を施す画像処理装置と、該画像処理の結果に基づいてドットを形成することにより出力媒体上に画像を出力する画像出力装置とを備える画像出力システムであって、

前記画像処理装置は、

前記画像を構成する複数の画素が所定個数ずつまとめられた画素群毎に、該画素群を代表する階調値たる画素群階調値を、該画素群内の各画素の画像データに基づいて決定する画素群階調値決定手段と、

前記画素群毎に付与された分類番号および該画素群の画素群階調値の組合せと、前記画素群に形成するドット個数を表す個数データとの対応関係たる第 1 の対応関係を参照することにより、前記各々の画素群についての個数データを生成して前記画像出力装置に供給する個数データ供給手段と

を備えており、

前記画像出力装置は、

前記画素群内の各画素について、該画素群内でドットが形成される順番を示す順序値を記憶している順序値記憶手段と、

前記順序値および前記個数データの組合せと、該順序値を有する画素におけるドット形成の有無との対応関係たる第 2 の対応関係を参照することにより、前記個数データを受け取った画素群内の各画素についてドット形成の有無を決定するドット形成有無決定手段と、

前記決定したドット形成の有無に従って前記出力媒体上にドットを形成するドット形成手段と

を備えている画像出力システム。

**【請求項 2】**

請求項 1 記載の画像出力システムであって、

前記画像処理装置は、前記画素群を前記画像中での位置に応じて複数種類に分類することにより、該各々の画素群に前記分類番号を付与する分類番号付与手段を備える画像出力システム。

**【請求項 3】**

画像データに所定の画像処理を施す第 1 の画像処理装置と、出力媒体上にドットを形成して画像を出力する際に該ドットの形成を制御するために用いられる制御データを、該画像処理の結果に基づいて生成する第 2 の画像処理装置とを備える画像処理システムであって、

前記第 1 の画像処理装置は、

前記画像を構成する複数の画素が所定個数ずつまとめられた画素群毎に、該画素群を代表する階調値たる画素群階調値を、該画素群内の各画素の画像データに基づいて決定する画素群階調値決定手段と、

前記画素群毎に付与された分類番号および該画素群の画素群階調値の組合せと、前記画素群に形成するドット個数を表す個数データとの対応関係たる第 1 の対応関係を参照することにより、前記各々の画素群についての個数データを生成して前記第 2 の画像処理装置に供給する個数データ供給手段と

を備えており、

前記第 2 の画像処理装置は、

前記画素群内の各画素について、該画素群内でドットが形成される順番を示す順序値を記憶している順序値記憶手段と、

前記順序値および前記個数データの組合せと、該順序値を有する画素におけるドット形成の有無との対応関係たる第 2 の対応関係を参照しながら、前記個数データを受け取った画素群内の各画素についてドット形成の有無を決定することにより、前記制御データを生成する制御データ生成手段と

を備えている画像処理システム。

【請求項 4】

請求項 3 記載の画像処理システムであって、

前記順序値記憶手段は、前記画素群内でドットが形成される画素の序列を複数組記憶しているとともに、該画素の序列毎に各画素についての前記順序値を記憶している手段であり、

前記制御データ生成手段は、前記複数組の序列の中から前記画素群毎に 1 の序列を選択し、選択した序列に記憶されている前記順序値を用いて、該画素群内の各画素についてのドット形成の有無を決定することにより、前記制御データを生成する手段である画像処理システム。

【請求項 5】

請求項 3 または請求項 4 記載の画像処理システムであって、

前記第 1 の画像処理装置は、前記画素群を前記画像中での位置に応じて複数種類に分類することにより、該各々の画素群に前記分類番号を付与する分類番号付与手段を備える画像処理システム。

【請求項 6】

請求項 5 記載の画像処理システムであって、

前記分類番号付与手段は、複数の閾値を二次元的に配列したディザマトリックスを前記画像に適用したときの、該ディザマトリックスに対する前記画素群の相対位置に基づいて、前記分類番号を付与する手段であり、

前記個数データ供給手段は、前記画素群内では全画素が前記画素群階調値を有するものとして、該画素群の各画素に前記ディザマトリックスを用いてディザ法を適用したときに該画素群内に形成されるドット個数のデータを、前記分類番号と該画素群階調値との組合せに対応付けて、前記第 1 の対応関係として記憶している手段であり、

前記順序値記憶手段は、前記個数データの生成に用いたディザマトリックスを複数の画素群に分割し、該画素群内の各画素に対応付けられた閾値の大小関係に基づいて画素群毎に決定された画素の序列を、前記複数組の画素の序列として記憶している手段であり、

前記制御データ生成手段は、前記画像上での画素群の位置に対応した 1 の序列を選択して、該画素群内の各画素についてのドット形成の有無を決定することにより、前記制御データを生成する手段である画像処理システム。

【請求項 7】

前記分類番号付与手段、前記個数データ供給手段、および前記順序値記憶手段で用いられている前記ディザマトリックスが、ブルーノイズマスク特性を有するマトリックスである請求項 6 記載の画像処理システム。

【請求項 8】

前記分類番号付与手段、前記個数データ供給手段、および前記順序値記憶手段で用いられている前記ディザマトリックスが、グリーンノイズマスク特性を有するマトリックスである請求項 6 記載の画像処理システム。

【請求項 9】

請求項 5 記載の画像処理システムであって、

前記分類番号付与手段は、画素の大きさが前記画素群の大きさと一致するように前記画像データの解像度を変更し、解像度の変更された画素の各々に、前記画像中での位置に応じて前記分類番号を付与する手段であり、

前記画素群階調値決定手段は、前記解像度の変更された画素についての前記画像データの階調値を、前記画素群階調値として決定する手段である画像処理システム。

【請求項 10】

請求項 3 または請求項 4 記載の画像処理システムであって、

前記個数データ供給手段は、表現する階調値の異なる複数種類のドットを前記画素群に形成するときの、各種ドットの個数の組合せを表す前記個数データと、前記分類番号および画素群階調値の組合せとの対応関係を、前記第 1 の対応関係として記憶している手段で



あり、

前記制御データ生成手段は、前記順序値および前記個数データの組合せと、該順序値を有する画素に形成されるドット種類との対応関係を、前記第 2 の対応関係として記憶している手段である画像処理システム。

【請求項 1 1】

請求項 3 または請求項 4 記載の画像処理システムであって、

前記画素群階調値決定手段は、互いに所定の位置関係にある 4 個ないし 1 6 個ずつの画素を前記画素群としてまとめて、前記画素群階調値を決定する手段である画像処理システム。

【請求項 1 2】

画像データに応じて出力媒体上にドットを形成することにより、該画像データに対応した画像を出力する画像出力装置であって、

前記画像を構成する複数の画素が所定個数ずつまとめられた画素群毎に、該画素群を代表する階調値たる画素群階調値を、該画素群内の各画素の画像データに基づいて決定する画素群階調値決定手段と、

前記画素群毎に付与された分類番号および該画素群の画素群階調値の組合せと、前記画素群に形成するドット個数を表す個数データとの対応関係たる第 1 の対応関係を参照することにより、前記各々の画素群についての個数データを生成して前記画像出力装置に供給する個数データ供給手段と

前記画素群内の各画素について、該画素群内でドットが形成される順番を示す順序値を記憶している順序値記憶手段と、

前記順序値および前記個数データの組合せと、該順序値を有する画素におけるドット形成の有無との対応関係たる第 2 の対応関係を参照することにより、前記個数データを生成した画素群内の各画素について、ドット形成の有無を決定するドット形成有無決定手段と

前記決定したドット形成の有無に従って前記出力媒体上にドットを形成するドット形成手段と

を備えた画像出力装置。

【請求項 1 3】

請求項 1 2 記載の画像出力装置であって、

前記画素群を前記画像中での位置に応じて複数種類に分類することにより、該各々の画素群に前記分類番号を付与する分類番号付与手段を備える画像出力装置。

【請求項 1 4】

ドットを形成して画像を出力する画像出力装置が該ドットの形成を制御するために用いる制御データを、該画像を表す画像データに所定の画像処理を加えることによって生成する画像処理装置であって、

前記画像を構成する複数の画素が所定個数ずつまとめられた画素群毎に、該画素群を代表する階調値たる画素群階調値を、該画素群内の各画素の画像データに基づいて決定する画素群階調値決定手段と、

前記画素群毎に付与された分類番号および該画素群の画素群階調値の組合せと、前記画素群に形成するドット個数を表す個数データとの対応関係たる第 1 の対応関係を参照することにより、前記各々の画素群についての個数データを生成して前記第 2 の画像処理装置に供給する個数データ供給手段と、

前記画素群内の各画素について、該画素群内でドットが形成される順番を示す順序値を記憶している順序値記憶手段と、

前記順序値および前記個数データの組合せと、該順序値を有する画素におけるドット形成の有無との対応関係たる第 2 の対応関係を参照しながら、前記個数データを生成した画素群内の各画素についてドット形成の有無を決定することにより、前記制御データを生成する制御データ生成手段と

を備える画像処理装置。

**【請求項 15】**

請求項 14 記載の画像処理装置であって、

前記画素群を前記画像中での位置に応じて複数種類に分類することにより、該各々の画素群に前記分類番号を付与する分類番号付与手段を備える画像出力装置。

**【請求項 16】**

画像データに応じて出力媒体上にドットを形成することにより、該画像データに対応した画像を出力する画像出力方法であって、

前記画像を構成する複数の画素が所定個数ずつまとめられた画素群毎に、該画素群を代表する階調値たる画素群階調値を、該画素群内の各画素の画像データに基づいて決定する第 1 の工程と、

前記画素群毎に付与された分類番号および該画素群の画素群階調値の組合せと、前記画素群に形成するドット個数を表す個数データとの対応関係たる第 1 の対応関係を参照することにより、前記各々の画素群についての個数データを生成する第 2 の工程と、

前記画素群内の各画素について、該画素群内でドットが形成される順番を示す順序値を記憶しておく第 3 の工程と、

前記順序値および前記個数データの組合せと、該順序値を有する画素におけるドット形成の有無との対応関係たる第 2 の対応関係を参照することにより、前記個数データを生成した画素群内の各画素について、ドット形成の有無を決定する第 4 の工程と、

前記決定したドット形成の有無に従って前記出力媒体上にドットを形成する第 5 の工程と

を備える画像出力方法。

**【請求項 17】**

ドットを形成して画像を出力する際に該ドットの形成を制御するために用いられる制御データを、該画像を表す画像データに所定の画像処理を加えることで生成する画像処理方法であって、

前記画像を構成する複数の画素が所定個数ずつまとめられた画素群毎に、該画素群を代表する階調値たる画素群階調値を、該画素群内の各画素の画像データに基づいて決定する工程 (A) と、

前記画素群毎に付与された分類番号および該画素群の画素群階調値の組合せと、前記画素群に形成するドット個数を表す個数データとの対応関係たる第 1 の対応関係を参照することにより、前記各々の画素群についての個数データを生成する工程 (B) と、

前記画素群内の各画素について、該画素群内でドットが形成される順番を示す順序値を記憶しておく工程 (C) と、

前記順序値および前記個数データの組合せと、該順序値を有する画素におけるドット形成の有無との対応関係たる第 2 の対応関係を参照しながら、前記個数データを生成した画素群内の各画素についてドット形成の有無を決定することにより、前記制御データを生成する工程 (D) と

を備える画像処理方法。

**【請求項 18】**

画像データに応じて出力媒体上にドットを形成することにより、該画像データに対応した画像を出力する方法を、コンピュータを用いて実現するためのプログラムであって、

前記画像を構成する複数の画素が所定個数ずつまとめられた画素群毎に、該画素群を代表する階調値たる画素群階調値を、該画素群内の各画素の画像データに基づいて決定する第 1 の機能と、

前記画素群毎に付与された分類番号および該画素群の画素群階調値の組合せと、前記画素群に形成するドット個数を表す個数データとの対応関係たる第 1 の対応関係を参照することにより、前記各々の画素群についての個数データを生成する第 2 の機能と、

前記画素群内の各画素について、該画素群内でドットが形成される順番を示す順序値を記憶しておく第 3 の機能と、

前記順序値および前記個数データの組合せと、該順序値を有する画素におけるドット形

成の有無との対応関係たる第2の対応関係を参照することにより、前記個数データを生成した画素群内の各画素について、ドット形成の有無を決定する第4の機能と、

前記決定したドット形成の有無に従って前記出力媒体上にドットを形成する第5の機能と

を実現するプログラム。

【請求項19】

ドットを形成して画像を出力する際に該ドットの形成を制御するために用いられる制御データを、該画像を表す画像データに所定の画像処理を加えることで生成する方法を、コンピュータを用いて実現するためのプログラムであって、

前記画像を構成する複数の画素が所定個数ずつまとめられた画素群毎に、該画素群を代表する階調値たる画素群階調値を、該画素群内の各画素の画像データに基づいて決定する機能（A）と、

前記画素群毎に付与された分類番号および該画素群の画素群階調値の組合せと、前記画素群に形成するドット個数を表す個数データとの対応関係たる第1の対応関係を参照することにより、前記各々の画素群についての個数データを生成する機能（B）と、

前記画素群内の各画素について、該画素群内でドットが形成される順番を示す順序値を記憶しておく機能（C）と、

前記順序値および前記個数データの組合せと、該順序値を有する画素におけるドット形成の有無との対応関係たる第2の対応関係を参照しながら、前記個数データを生成した画素群内の各画素についてドット形成の有無を決定することにより、前記制御データを生成する機能（D）と

を実現するプログラム。

**【書類名】明細書**

**【発明の名称】** 所定領域内に形成されるドット個数の情報に基づいて画像を出力する画像出力システム

**【技術分野】****【0001】**

この発明は、画像データに基づいて画像を出力する技術に関し、詳しくは、画像データに所定の画像処理を施してドットを適切な密度で発生させることにより、画像を出力する技術に関する。

**【背景技術】****【0002】**

印刷媒体や液晶画面といった各種の出力媒体上にドットを形成することで画像を出力する画像出力装置は、各種画像機器の出力装置として広く使用されている。これら画像出力装置では、画像は画素と呼ばれる小さな領域に細分された状態で扱われており、ドットはこれら画素に形成される。画素にドットを形成する場合、もちろん画素1つ1つについて見れば、ドットが形成されるか否かのいずれかの状態しか取り得ない。しかし、ある程度の広さを持った領域で見れば、形成するドットの密度に粗密を生じさせることが可能であり、ドットの形成密度を変えることによって多階調の画像を出力することが可能である。例えば、印刷用紙上に黒いインクのドットを形成する場合、ドットが密に形成されている領域は暗く見えるし、逆にドットがまばらに形成されている領域は明るく見える。また、液晶画面に輝点のドットを形成する場合、ドットが密に形成された領域は明るく見え、まばらに形成された領域は暗く見える。従って、ドットの形成密度を適切に制御してやれば、多階調の画像を出力することが可能となる。このように適切な形成密度が得られるようにドットの形成を制御するためのデータは、出力しようとする画像に所定の画像処理を施すことによって発生させる。

**【0003】**

近年では、これら画像出力装置には、出力画像の高画質化や大画像化が要請されるようになってきた。高画質化の要請に対しては、画像をより細かな画素に分割することが効果的である。画素を小さくしてやれば、画素に形成されるドットが目立たなくなるので画質を向上させることができる。また、大画像化の要請に対しては、画素数の増加によって対応する。もちろん、個々の画素を大きくすることによっても出力画像を大きくすることはできるが、これでは画質の低下を招いてしまうので、大型化の要請に対しては画素数を増加させることが効果的である。

**【0004】**

もっとも、画像を構成する画素数が増加すると画像処理に時間がかかってしまい、画像を迅速に出力することが困難となる。そこで、画像処理を迅速に実行可能とする技術が提案されている（特許文献1）。

**【0005】**

**【特許文献1】** 特開2002-185789号公報

**【発明の開示】****【発明が解決しようとする課題】****【0006】**

しかし、画像処理を迅速に行ったとしても、画像データの転送や、処理済みの画像データの転送に時間がかかってしまったり、あるいは転送された画像データの処理に時間がかかってしまったのでは、画像の出力を迅速化する効果にも自ずから限界がある。

**【0007】**

また、近年では、デジタルカメラなどで撮影した画像のデータを、印刷装置などの画像出力装置に直接供給して直ちに画像を出力したいという要請も存在する。このような場合は、いわゆるパーソナルコンピュータ等のような高い処理能力を備えた画像処理装置を用いて画像処理を行うことはできない。従って、デジタルカメラなどの画像撮影装置あるいは画像出力装置のいずれか、若しくは両方で分担して実行可能なように、簡素な画像処理

としておく必要がある。

【0008】

この発明は従来技術における上述した課題を解決するためになされたものであり、十分な出力画質を維持したまま、画像データを迅速に処理可能であるとともに、画像処理のためにパーソナルコンピュータ等のような高い処理能力を有する機器を用いずとも実行可能な、簡素な画像処理技術の提供を目的とする。

【課題を解決するための手段】

【0009】

上述した課題の少なくとも一部を解決するために、本発明の画像出力システムは、次の構成を採用した。すなわち、

画像データに所定の画像処理を施す画像処理装置と、該画像処理の結果に基づいてドットを形成することにより出力媒体上に画像を出力する画像出力装置とを備える画像出力システムであって、

前記画像処理装置は、

前記画像を構成する複数の画素が所定個数ずつまとめられた画素群毎に、該画素群を代表する階調値たる画素群階調値を、該画素群内の各画素の画像データに基づいて決定する画素群階調値決定手段と、

前記画素群毎に付与された分類番号および該画素群の画素群階調値の組合せと、前記画素群に形成するドット個数を表す個数データとの対応関係たる第1の対応関係を参照することにより、前記各々の画素群についての個数データを生成して前記画像出力装置に供給する個数データ供給手段と

を備えており、

前記画像出力装置は、

前記画素群内の各画素について、該画素群内でドットが形成される順番を示す順序値を記憶している順序値記憶手段と、

前記順序値および前記個数データの組合せと、該順序値を有する画素におけるドット形成の有無との対応関係たる第2の対応関係を参照することにより、前記個数データを受け取った画素群内の各画素についてドット形成の有無を決定するドット形成有無決定手段と、

前記決定したドット形成の有無に従って前記出力媒体上にドットを形成するドット形成手段と

を備えていることを要旨とする。

【0010】

また、上記の画像出力システムに対応する本発明の画像出力方法は、

画像データに応じて出力媒体上にドットを形成することにより、該画像データに対応した画像を出力する画像出力方法であって、

前記画像を構成する複数の画素が所定個数ずつまとめられた画素群毎に、該画素群を代表する階調値たる画素群階調値を、該画素群内の各画素の画像データに基づいて決定する第1の工程と、

前記画素群毎に付与された分類番号および該画素群の画素群階調値の組合せと、前記画素群に形成するドット個数を表す個数データとの対応関係たる第1の対応関係を参照することにより、前記各々の画素群についての個数データを生成する第2の工程と、

前記画素群内の各画素について、該画素群内でドットが形成される順番を示す順序値を記憶しておく第3の工程と、

前記順序値および前記個数データの組合せと、該順序値を有する画素におけるドット形成の有無との対応関係たる第2の対応関係を参照することにより、前記個数データを生成した画素群内の各画素について、ドット形成の有無を決定する第4の工程と、

前記決定したドット形成の有無に従って前記出力媒体上にドットを形成する第5の工程と

を備えていることを要旨とする。

## 【0011】

かかる本発明の画像出力システムおよび画像出力方法においては、画像を複数の画素群に分割し、画素群内に形成されるドット個数を表す個数データを生成して、画像出力装置に供給する。画像出力装置は、供給された個数データに基づいて、画素群内の各画素についてドット形成の有無を決定した後、決定の結果に従って出力媒体上にドットを形成することによって画像を出力する。

## 【0012】

詳細には後述するが、画像の全画素についてドット形成の有無を表すデータに比べれば、画素群毎にドットの個数を表すデータは遙かに小さなデータとすることができる。このため、画像出力装置に個数データを供給してやれば、迅速にデータを供給することが可能となり、延いては画像を迅速に出力することが可能となる。

## 【0013】

また、個数データの生成に際しては、画素群の分類番号および画素群階調値の組合せと、個数データとの対応関係（第1の対応関係）を参照することによって個数データを生成する。対応関係を参照して個数データを生成する処理は極めて簡単な処理である。加えて、後ほど詳細に説明するように、画素群の画素群階調値は極めて容易に求めることができる。また、必要な場合は分類番号も極めて容易に決定することができるから、結局、個数データを生成する処理は極めて簡素な処理とすることができる。このため、個数データを迅速に生成することが可能となり、延いては画像出力装置に迅速に供給して、画像を速やかに出力することが可能となる。

## 【0014】

更に、画像出力装置では、次のようにして、画素群内の各画素についてドット形成の有無を決定する。まず、画素群内でドットが形成される順番を示す順序値を、画素群内の各画素について予め記憶しておく。そして、順序値および個数データの組合せと、該順序値を有する画素におけるドット形成の有無との対応関係（第2の対応関係）を参照することによって、個数データが供給された画素群内の各画素について、ドット形成の有無を決定する。このように対応関係を参照しながらドット形成の有無を決定してやれば、画素群内の各画素についてのドット形成の有無を、個数データから直ちに決定することができるので、極めて迅速に且つ簡便に決定することが可能であり、延いては画像を迅速に出力することが可能となる。

## 【0015】

更に、個数データを生成する処理も、個数データから画素群内の各画素についてのドット形成有無を決定する処理も、このように極めて簡素な処理で実行することができれば、コンピュータなどのように高度な処理能力を有さない機器においても、十分に実用的な速度で処理することが可能である。従って、例えば画像データを、コンピュータなどを介さずに直接、画像出力装置に供給し、画像データにこれらの画像処理を画像出力装置の内部で施して、適切に画像を出力することも可能となる。

## 【0016】

また、画素群の分類番号は、各画素群を画像中での位置に応じて複数種類に分類することによって付与することとしても良い。こうすれば、画素群に予め分類番号を付与しておかずとも、必要に応じて適宜、分類番号を付与することができる。また、画像中での位置に応じて付与することで、分類番号を適切に付与することが可能となる。

## 【0017】

このような観点からは、本発明を次のような画像出力装置として把握することも可能である。すなわち、上記の画像出力システム、あるいは画像出力方法に対応する本発明の画像出力装置は、

画像データに応じて出力媒体上にドットを形成することにより、該画像データに対応した画像を出力する画像出力装置であって、

前記画像を構成する複数の画素が所定個数ずつまとめられた画素群毎に、該画素群を代表する階調値たる画素群階調値を、該画素群内の各画素の画像データに基づいて決定する

画素群階調値決定手段と、

前記画素群毎に付与された分類番号および該画素群の画素群階調値の組合せと、前記画素群に形成するドット個数を表す個数データとの対応関係たる第1の対応関係を参照することにより、前記各々の画素群についての個数データを生成して前記画像出力装置に供給する個数データ供給手段と

前記画素群内の各画素について、該画素群内でドットが形成される順番を示す順序値を記憶している順序値記憶手段と、

前記順序値および前記個数データの組合せと、該順序値を有する画素におけるドット形成の有無との対応関係たる第2の対応関係を参照することにより、前記個数データを生成した画素群内の各画素について、ドット形成の有無を決定するドット形成有無決定手段と

前記決定したドット形成の有無に従って前記出力媒体上にドットを形成するドット形成手段と

を備えることを要旨とする。

#### 【0018】

かかる画像出力装置においては、画像を複数の画素群に分割し、第1の対応関係を参照することによって画素群毎に個数データを生成する。次いで、第2の対応関係を参照することによって、個数データから画素群内の各画素についてのドット形成の有無を決定する。こうして決定した結果に基づいて出力媒体上にドットを形成することによって画像を出力する。このように、これら対応関係を参照しながら画像データを変換してやれば、簡便に且つ迅速に画像を出力することが可能である。更に、コンピュータなどのようには高度な処理能力を有さない機器においても、十分に実用的な速度で画像を出力可能な画像出力装置を構成することが可能となる。

#### 【0019】

また、前述した課題の少なくとも一部を解決するために、本発明の画像処理システムは、次の構成を採用した。すなわち、

画像データに所定の画像処理を施す第1の画像処理装置と、出力媒体上にドットを形成して画像を出力する際に該ドットの形成を制御するために用いられる制御データを、該画像処理の結果に基づいて生成する第2の画像処理装置とを備える画像処理システムであって、

前記第1の画像処理装置は、

前記画像を構成する複数の画素が所定個数ずつまとめられた画素群毎に、該画素群を代表する階調値たる画素群階調値を、該画素群内の各画素の画像データに基づいて決定する画素群階調値決定手段と、

前記画素群毎に付与された分類番号および該画素群の画素群階調値の組合せと、前記画素群に形成するドット個数を表す個数データとの対応関係たる第1の対応関係を参照することにより、前記各々の画素群についての個数データを生成して前記第2の画像処理装置に供給する個数データ供給手段と

を備えており、

前記第2の画像処理装置は、

前記画素群内の各画素について、該画素群内でドットが形成される順番を示す順序値を記憶している順序値記憶手段と、

前記順序値および前記個数データの組合せと、該順序値を有する画素におけるドット形成の有無との対応関係たる第2の対応関係を参照しながら、前記個数データを受け取った画素群内の各画素についてドット形成の有無を決定することにより、前記制御データを生成する制御データ生成手段と

を備えていることを要旨とする。

#### 【0020】

また、上記の画像処理システムに対応する本発明の画像処理方法は、

ドットを形成して画像を出力する際に該ドットの形成を制御するために用いられる制御



データを、該画像を表す画像データに所定の画像処理を加えることで生成する画像処理方法であって、

前記画像を構成する複数の画素が所定個数ずつまとめられた画素群毎に、該画素群を代表する階調値たる画素群階調値を、該画素群内の各画素の画像データに基づいて決定する工程（A）と、

前記画素群毎に付与された分類番号および該画素群の画素群階調値の組合せと、前記画素群に形成するドット個数を表す個数データとの対応関係たる第1の対応関係を参照することにより、前記各々の画素群についての個数データを生成する工程（B）と、

前記画素群内の各画素について、該画素群内でドットが形成される順番を示す順序値を記憶しておく工程（C）と、

前記順序値および前記個数データの組合せと、該順序値を有する画素におけるドット形成の有無との対応関係たる第2の対応関係を参照しながら、前記個数データを生成した画素群内の各画素についてドット形成の有無を決定することにより、前記制御データを生成する工程（D）と

を備えることを要旨とする。

#### 【0021】

かかる本発明の画像処理システムおよび画像処理方法においても、互いに隣接する画素を所定数ずつまとめて画素群を形成することで、画像を複数の画素群に分割し、画素群内に形成されるドット個数を表す個数データを生成する。次いで、画素群毎に生成した個数データに基づいて、画素群内の各画素についてドット形成の有無を決定することにより、制御データを生成する。

#### 【0022】

後述するように、画像の全画素についてドット形成の有無を表すデータに比べれば、画素群毎にドットの個数を表すデータは小さなデータであるため、データの取り扱いが容易となり、その分だけ簡便に且つ迅速に制御データを生成することが可能となる。

#### 【0023】

また、個数データを生成するに際しては、第1の対応関係、すなわち、画素群の分類番号および画素群階調値の組合せと、個数データとの対応関係を参照することによって生成しているため、迅速に且つ極めて簡単な処理で個数データを生成することができる。更に、個数データから制御データを生成するに際しても、第2の対応関係、すなわち、順序値および個数データの組合せと、該順序値を有する画素におけるドット形成の有無との対応関係を参照することによって生成しているため、迅速に且つ極めて簡単な処理によって、個数データから制御データを生成することができる。結局、画像データから制御データを簡便且つ迅速に生成することができ、延いては生成した制御データを用いることによって画像を迅速に出力することが可能となる。

#### 【0024】

更に、画像データから制御データを極めて簡単な処理で生成することができるので、コンピュータなどのように高度な処理能力を有さない機器においても、十分に実用的な速度で制御データを生成することが可能である。

#### 【0025】

このような観点からは、本発明を次のような画像処理装置として把握することも可能である。すなわち、上記の画像処理システム、あるいは画像処理方法に対応する本発明の画像処理装置は、

ドットを形成して画像を出力する画像出力装置が該ドットの形成を制御するために用いる制御データを、該画像を表す画像データに所定の画像処理を加えることによって生成する画像処理装置であって、

前記画像を構成する複数の画素が所定個数ずつまとめられた画素群毎に、該画素群を代表する階調値たる画素群階調値を、該画素群内の各画素の画像データに基づいて決定する画素群階調値決定手段と、

前記画素群毎に付与された分類番号および該画素群の画素群階調値の組合せと、前記画



素群に形成するドット個数を表す個数データとの対応関係たる第1の対応関係を参照することにより、前記各々の画素群についての個数データを生成して前記第2の画像処理装置に供給する個数データ供給手段と、

前記画素群内の各画素について、該画素群内でドットが形成される順番を示す順序値を記憶している順序値記憶手段と、

前記順序値および前記個数データの組合せと、該順序値を有する画素におけるドット形成の有無との対応関係たる第2の対応関係を参照しながら、前記個数データを生成した画素群内の各画素についてドット形成の有無を決定することにより、前記制御データを生成する制御データ生成手段と

を備えることを要旨とする。

#### 【0026】

かかる画像処理装置においては、画像を複数の画素群に分割した後、第1の対応関係を参照することによって画像データから画素群毎に個数データを生成し、次いで、第2の対応関係を参照することによって、画素群毎の個数データから制御データを生成する。このように、これら対応関係を参照しながら変換すれば、簡便に且つ迅速に制御データを生成することが可能である。更に、コンピュータなどのようには高度な処理能力を有さない機器においても、十分に実用的な速度で制御データを生成可能な画像処理装置を構成することが可能となる。

#### 【0027】

こうした画像処理システム、画像処理方法、あるいは画像処理装置においては、次のようにしても良い。まず、画素群内でドットが形成される画素の序列を複数組記憶しておく、それぞれの画素の序列毎に、画素群内の各画素についての順序値を記憶しておく。そして、複数組の序列の中から画素群毎に1の序列を選択し、選択した序列に記憶されている順序値を用いて、画素群内の各画素についてのドット形成の有無を決定することにより、制御データを生成する。

#### 【0028】

画素群内の各画素についてのドット形成の有無は、その画素群の個数データと該画素群内の各画素の順序値とに基づいて決定される。ここで、複数組の序列の中から、画素群毎に1の序列を選択しながらドット形成の有無を決定してやれば、各画素の順序値は画素群毎に異なったものとなる。従って、複数の画素群に亘って、同じ個数データが続いた場合でも、順序値が異なっているために、これら画素群で同じ画素位置にドットが形成されることがない。このため、同じパターンでドットが形成された領域が目立ってしまい、画質が悪化することを確実に回避することが可能となる。

#### 【0029】

ここで、画素群の分類番号は、各画素群を画像中での位置に応じて複数種類に分類することによって付与することとしても良い。こうすれば、画素群に予め分類番号を付与しておかずとも、必要に応じて適宜、分類番号を付与することができる。また、画像中での位置に応じて付与することで、分類番号を適切に付与することが可能となる。

#### 【0030】

また、こうした画像処理システム、画像処理方法、あるいは画像処理装置においては、複数の閾値が二次元的に配列されたディザマトリックスを想定し、このディザマトリックスに基づいて設定された分類番号と個数データと順序値とを用いて、制御データを生成することとしてもよい。一例として、ある画素群に着目して説明する。まず、画像にディザマトリックスを適用したときの、マトリックスに対する画素群の相対位置に基づいて、画素群の分類番号を付与してやる。次に、その画素群内では全画素が画素群階調値を有するものとして、ディザマトリックスを用いてディザ法を適用することにより、該画素群内に形成されるドットの個数を求める。こうして求めたドットの個数を表す個数データを、分類番号および画素群階調値の組合せと対応付けて、第1の対応関係として記憶しておく。更に、ディザマトリックスを画像に適用したときに、画素群に対応する領域に設定されている閾値の大きさに従って、その画素群内の各画素についての順序値を決定し、得られた

順序値の組を複数組の画素の序列として記憶しておく。そして、画素群の個数データを生成したら、画像上で該画素群の位置に対応した1の序列を選択して、該序列に設定されている順序値を用いて各画素についてのドット形成の有無を判断することにより、制御データを生成する。

#### 【0031】

詳細には後述するが、このように分類番号と個数データと順序値とを、同じディザマトリックスに基づいて設定しておけば、ディザ法を用いて画素毎にドット形成の有無を判断した場合と全く同等の画質で画像を出力することが可能となる。特に、画素群としてまとめられる画素の画像データが同じ階調値を有している場合は、ディザ法を用いて画素毎にドット形成の有無を判断した場合と、個数データから各画素についてのドット形成の有無を決定した場合とでは、全く同じ画素位置にドットが形成されることになる。

#### 【0032】

ここで、ディザ法を用いた場合と全く同じ画素位置にドットが形成されることから明らかなように、分類番号、個数データ、順序値を、ディザマトリックスを用いて設定しておくことで、ドットの発生状況を比較的自由に制御することが可能である。すなわち、ドットの発生状況は、画像データを画素群単位で処理しているにも関わらず画素群の大きさの影響はほとんど受けず、分類番号や順序値などを設定するために用いたディザマトリックスに大きく依存する。そして、特に、画素群としてまとめられる画素の画像データが同じ階調値を有している場合は、ドットの発生状況は、ディザマトリックスによって完全に決定されることになる。

#### 【0033】

こうした特性に鑑みて、分類番号、個数データ、順序値を設定するに際しては、いわゆるブルーノイズマスク特性を有するディザマトリックスに基づいて設定することとしても良い。ここで、本明細書における「ブルーノイズマスク特性を有するディザマトリックス」とは、次のようなマトリックスを言う。すなわち、ドットを不規則に発生させるとともに、設定されている閾値の空間周波数成分は、1周期が2画素以下の高周波数領域に最も大きな成分を有するディザマトリックスを言う。尚、明るい（明度の高い）画像など、特定の明度付近では規則的なパターンでドットが形成される場合があっても良い。

#### 【0034】

分類番号、個数データ、順序値を、このようなブルーノイズマスク特性を有するディザマトリックスに基づいて設定しておけば、ブルーノイズマスク特性が反映されたドット分布で画像を出力することができる。すなわち、個数データの生成とドット形成の有無とを画素群単位で行っているにも関わらず、ドットの分布には画素群に相当する周期的なパターンが発生することがない。それでいて、画素群単位で個数データを扱うことにより、画像を迅速に出力することが可能であり、従って、高画質な画像を迅速に出力することができる。

#### 【0035】

あるいは、分類番号、個数データ、順序値を設定するに際しては、いわゆるグリーンノイズマスク特性を有するディザマトリックスに基づいて設定することとしても良い。ここで、本明細書における「グリーンノイズマスク特性を有するディザマトリックス」とは、次のようなマトリックスを言う。すなわち、ドットを不規則に発生させるとともに、設定されている閾値の空間周波数成分は、1周期が2画素から十数画素の中間周波数領域に最も大きな成分を有するディザマトリックスを言う。尚、ここでも、特定の明度付近であれば規則的なパターンでドットが形成される場合があっても良い。

#### 【0036】

いわゆるレーザープリンタなどのように、1画素程度の微細なドットを安定して形成することが苦手な画像出力機器では、こうしたグリーンノイズマスク特性を有するディザマトリックスを用いることにより、孤立したドットの発生を抑制することができる。その結果、安定した画質の画像を迅速に出力することが可能となる。

#### 【0037】

こうした画像処理システム、画像処理方法、あるいは画像処理装置においては、画像データの解像度を次のような解像度に調整しておくことにより、明示的には所定数の画素をまとめて画素群を形成することなく、個数データを生成することとしてもよい。すなわち、変換後の画素の大きさが画素群の大きさと一致するような解像度に、画像データを変更する。次いで、解像度が調整された画素の1つ1つを画素群として取り扱うことで分類番号を付与するとともに、各画素についての画像データの階調値を画素群階調値として取り扱うことにより、1つ1つの画素について個数データを生成する。こうして明示的には複数の画素を画素群にまとめることなく、画素群の個数データを生成する。

#### 【0038】

画質上の要請から、画像データの解像度よりも高い解像度で画像を印刷することがしばしば行われる。このような場合に、上述した方法で個数データを生成してやれば、画像データを、印刷しようとする解像度よりも低い解像度に変換して個数データを生成することができる。画像データは一般に解像度が高くなるほどデータ量が多くなって取り扱いが困難となることから、より低い解像度のままで個数データを生成することで、データの取り扱いが容易になるとともに、個数データを生成する処理も迅速化することが可能となる。

#### 【0039】

また、上述した画像処理システム、画像処理方法、あるいは画像処理装置においては、画素群内に形成されるドット個数を表す個数データとして、表現する階調値が互いに異なる複数種類のドットについてのドット個数の組合せを表すデータを生成することとしてもよい。ここで、表現する階調値が互いに異なる複数種類のドットとは、例えば、ドットの大きさが異なる複数種類のドットとすることもできるし、あるいは、ドットの濃度が異なる複数種類のドットとすることもできる。更には、微細なドットを所定の密度で形成することで擬似的に1つのドットを形成している場合には、微細なドットの密度が異なった複数種類のドットとすることも可能である。

#### 【0040】

このような各種ドットの個数の組合せを表す個数データと、分類番号および画素群階調値の組合せとの対応関係を示す第1の対応関係を参照することによって、画素群毎の個数データを生成する。そして、順序値および個数データの組合せと、該順序値を有する画素に形成されるドット種類との対応関係を示す第2の対応関係を参照することによって、画素群毎に生成された個数データから、各画素について各種ドットの形成の有無を決定することにより、制御データを生成する。

#### 【0041】

第1の対応関係を参照することによって個数データを生成する場合、個数データが複数種類のドットについてのドット個数の組合せを示すデータであっても、単にドット個数を表すデータである場合と同様に、極めて容易に個数データを生成することができる。通常、画素毎にドット形成の有無を判断する場合は、ドットの種類が多くなると、それに連れて判断する処理もいきおい複雑なものとなり勝ちであるから、第1の対応関係を参照して個数データを生成することで、ドットの種類が多くなるほど、相対的に迅速に生成することが可能となるので好ましい。また、画素毎にドット形成の有無を決定する処理も、ドットの種類が多くなるほど複雑なものとなりがちであるが、第2の対応関係を参照しながら各種ドットについての形成の有無を決定してやれば、ドットの種類が多くなっても簡便に決定することができる。すなわち、相対的には迅速にドット形成の有無を決定することができる。結局、ドットの種類が多くなるほど、画像データから制御データを迅速に生成することが可能となるので好ましい。

#### 【0042】

更に、上述した画像処理システム、画像処理方法、あるいは画像処理装置においては、互いに所定の位置関係にある4個ないし16個ずつの画素を画素群としてまとめて、画素群階調値を決定することとしても良い。

#### 【0043】

画素群としてまとめる画素数が少なくなるほど、画素群の個数が増加するので、第1の

対応関係は複雑なものとなる。従って、この観点からは、画素群にまとめる画素数は多い方が好ましい。一方、画素群に含まれる画素の階調値は画素群階調値にまとめられてしまうから、画素群にまとめる画素数があまりに多くなったのでは画質の悪化を引き起こすおそれが生じる。こうした点を踏まえると、経験上、最も良好な結果を得ることができるのは、8画素から16画素ずつの画素を画素群としてまとめた場合であるが、4画素から16画素ずつを画素群としてまとめた場合でも十分な結果を得ることができる。また、画素群にまとめられる複数の画素の位置関係は、例えば、主走査方向に4画素ずつ、副走査方向に2画素ずつといったように、互いに矩形形状を成すような位置関係としておけば、経験上、良好な画質を得ることが可能である。

#### 【0044】

更に本発明は、上述した画像出力方法あるいは画像処理方法を実現するためのプログラムをコンピュータに読み込ませ、コンピュータを用いて実現することも可能である。従って、本発明は次のようなプログラム、あるいは該プログラムを記録した記録媒体としての態様も含んでいる。すなわち、上述した画像出力方法に対応する本発明のプログラムは、

画像データに応じて出力媒体上にドットを形成することにより、該画像データに対応した画像を出力する方法を、コンピュータを用いて実現するためのプログラムであって、

前記画像を構成する複数の画素が所定個数ずつまとめられた画素群毎に、該画素群を代表する階調値たる画素群階調値を、該画素群内の各画素の画像データに基づいて決定する第1の機能と、

前記画素群毎に付与された分類番号および該画素群の画素群階調値の組合せと、前記画素群に形成するドット個数を表す個数データとの対応関係たる第1の対応関係を参照することにより、前記各々の画素群についての個数データを生成する第2の機能と、

前記画素群内の各画素について、該画素群内でドットが形成される順番を示す順序値を記憶しておく第3の機能と、

前記順序値および前記個数データの組合せと、該順序値を有する画素におけるドット形成の有無との対応関係たる第2の対応関係を参照することにより、前記個数データを生成した画素群内の各画素について、ドット形成の有無を決定する第4の機能と、

前記決定したドット形成の有無に従って前記出力媒体上にドットを形成する第5の機能と

を実現することを要旨とする。

#### 【0045】

また、上記のプログラムに対応する本発明の記録媒体は、

画像データに応じて出力媒体上にドットを形成することにより、該画像データに対応した画像を出力する方法を、コンピュータを用いて実現するプログラムを、コンピュータで読み取り可能に記録した記録媒体であって、

前記画像を構成する複数の画素が所定個数ずつまとめられた画素群毎に、該画素群を代表する階調値たる画素群階調値を、該画素群内の各画素の画像データに基づいて決定する第1の機能と、

前記画素群毎に付与された分類番号および該画素群の画素群階調値の組合せと、前記画素群に形成するドット個数を表す個数データとの対応関係たる第1の対応関係を参照することにより、前記各々の画素群についての個数データを生成する第2の機能と、

前記画素群内の各画素について、該画素群内でドットが形成される順番を示す順序値を記憶しておく第3の機能と、

前記順序値および前記個数データの組合せと、該順序値を有する画素におけるドット形成の有無との対応関係たる第2の対応関係を参照することにより、前記個数データを生成した画素群内の各画素について、ドット形成の有無を決定する第4の機能と、

前記決定したドット形成の有無に従って前記出力媒体上にドットを形成する第5の機能と

をコンピュータを用いて実現するプログラムを記録していることを要旨とする。

#### 【0046】

更に、上述した画像処理方法に対応する本発明のプログラムは、

ドットを形成して画像を出力する際に該ドットの形成を制御するために用いられる制御データを、該画像を表す画像データに所定の画像処理を加えることで生成する方法を、コンピュータを用いて実現するためのプログラムであって、

前記画像を構成する複数の画素が所定個数ずつまとめられた画素群毎に、該画素群を代表する階調値たる画素群階調値を、該画素群内の各画素の画像データに基づいて決定する機能（A）と、

前記画素群毎に付与された分類番号および該画素群の画素群階調値の組合せと、前記画素群に形成するドット個数を表す個数データとの対応関係たる第1の対応関係を参照することにより、前記各々の画素群についての個数データを生成する機能（B）と、

前記画素群内の各画素について、該画素群内でドットが形成される順番を示す順序値を記憶しておく機能（C）と、

前記順序値および前記個数データの組合せと、該順序値を有する画素におけるドット形成の有無との対応関係たる第2の対応関係を参照しながら、前記個数データを生成した画素群内の各画素についてドット形成の有無を決定することにより、前記制御データを生成する機能（D）と

を実現することを要旨とする。

#### 【0047】

また、上記のプログラムに対応する本発明の記録媒体は、

ドットを形成して画像を出力する際に該ドットの形成を制御するために用いられる制御データを、該画像を表す画像データに所定の画像処理を加えることで生成するプログラムを、コンピュータで読み取り可能に記録した記録媒体であって、

前記画像を構成する複数の画素が所定個数ずつまとめられた画素群毎に、該画素群を代表する階調値たる画素群階調値を、該画素群内の各画素の画像データに基づいて決定する機能（A）と、

前記画素群毎に付与された分類番号および該画素群の画素群階調値の組合せと、前記画素群に形成するドット個数を表す個数データとの対応関係たる第1の対応関係を参照することにより、前記各々の画素群についての個数データを生成する機能（B）と、

前記画素群内の各画素について、該画素群内でドットが形成される順番を示す順序値を記憶しておく機能（C）と、

前記順序値および前記個数データの組合せと、該順序値を有する画素におけるドット形成の有無との対応関係たる第2の対応関係を参照しながら、前記個数データを生成した画素群内の各画素についてドット形成の有無を決定することにより、前記制御データを生成する機能（D）と

をコンピュータを用いて実現するプログラムを記録していることを要旨とする。

#### 【0048】

こうしたプログラム、あるいは記録媒体に記録されているプログラムをコンピュータに読み込ませ、該コンピュータを用いて上述の各種機能を実現させれば、十分な出力画質を維持したまま、画像データを迅速に処理可能であるとともに、画像処理のためにパーソナルコンピュータ等のような高い処理能力を有する機器を用いずとも実行可能な、簡素な画像処理を実現することが可能となる。

【発明を実施するための最良の形態】

#### 【0049】

以下では、本発明の作用・効果をより明確に説明するために、本発明の実施の形態を、次のような順序に従って説明する。

A. 発明の概要：

B. 第1実施例：

B-1. 装置構成：

B-2. 第1実施例の画像印刷処理の概要：

B-3. 個数データから画素位置を決定可能な原理：

- B-4. 第1実施例の個数データ生成処理:
- B-5. 分類番号の決定方法:
- B-6. 第1実施例のドット形成有無決定処理:
- B-7. 第1の変形例:
- B-8. 第2の変形例:
- B-9. メモリ使用量:
- C. 第2実施例:
  - C-1. 第2実施例の画像印刷処理:
  - C-2. 第2実施例の個数データ生成処理:
    - C-2-1. デイザ法を用いた大中小ドットの形成個数の決定処理:
    - C-2-2. 大中小ドットの形成個数のコード化処理:
    - C-2-3. 変換テーブルを利用した第2実施例の個数データ生成処理:
  - C-3. 第2実施例のドット形成有無決定処理:
    - C-3-1. 変換テーブルを参照せずにドット形成有無を決定する方法:
    - C-3-2. 変換テーブルを参照するドット形成有無決定処理:
  - C-4. メモリ使用量:

**【0050】****A. 発明の概要 :**

実施例についての詳細な説明に入る前に、図1を参照しながら、本発明の概要について説明しておく。図1は、印刷システムを例にとって、本発明の概要を説明するための説明図である。本印刷システムは、画像処理装置としてのコンピュータ10と、画像出力装置としてのプリンタ20等から構成されており、コンピュータ10に所定のプログラムがロードされて実行されると、コンピュータ10およびプリンタ20などが全体として、一体の画像出力システムとして機能する。プリンタ20は、印刷媒体上にドットを形成することによって画像を印刷する。コンピュータ10は、印刷しようとする画像の画像データに所定の画像処理を施すことによって、プリンタ20が画素毎にドットの形成を制御するためのデータを生成して、該プリンタ20に供給する。

**【0051】**

一般的な印刷システムでは、次のようにして画像を印刷する。まず、コンピュータで所定の画像処理を施すことにより、画像データを、画素毎にドット形成の有無を表すデータに変換する。次いで、得られたデータをプリンタに供給し、プリンタでは供給されたデータに従ってドットを形成することによって画像を印刷している。ここで、印刷しようとする画像の画素数が多くなると、それに伴って、画像処理に要する時間が増加し、画像を迅速に印刷することが困難となる。また、画素数が多くなるとつれて、画素毎にドット形成の有無を表すデータのデータ量が増加するので、コンピュータからプリンタにデータを出力するために要する時間が長くなるとともに、プリンタ側で受け取ったデータを処理するための時間も長くなり、それだけ印刷に要する時間が増加してしまう。

**【0052】**

こうした点に鑑みて、図1に例示した印刷システムでは、次のようにして画像を印刷する。まず、コンピュータ10では、画像を構成する画素を互いに隣接した所定個数ずつ画素群としてまとめることで、画像を複数の画素群に分割する。そして、各画素群について、画素群内に形成するドット個数を表す個数データを生成してプリンタ20に供給する。プリンタ20に供給された個数データは、ドット形成有無決定モジュールによって処理されることにより、画素群内の各画素についてドット形成の有無を表すデータに変換される。次いで、各画素について決定されたドット形成の有無に従って、ドット形成モジュールが印刷媒体上にドットを形成することによって画像が印刷される。

**【0053】**

ここで、画素毎にドット形成の有無を表すデータに比べれば、画素群毎の個数データは遙かに小さなデータ量とすることができる。従って、コンピュータ10から画素毎にドット形成の有無を表したデータをプリンタ20に供給する代わりに、画素毎の個数データを



供給してやれば、極めて迅速にデータを転送することが可能となる。

#### 【0054】

また、画素群の個数データは、コンピュータ10内で次のようにして生成される。まず、画素群階調値決定モジュールにおいて、画像を分割する複数の画素群について画素群階調値が決定される。画素群階調値とは、画素群を代表する階調値であり、該画素群内に含まれる各画素の画像データに基づいて決定される。また、第1対応関係記憶モジュールには、画素群に付与された分類番号と画素群階調値との組合せと、該組合せを有する画素群の個数データとの対応関係が、第1の対応関係として記憶されている。ここで画素群の分類番号は、各画素群を画像中での位置に応じて複数種類の分類することによって設定することもできるし、また、画像がいつも同じように分割される場合などには、各画素群毎に予め適切な分類番号を付与しておくこともできる。更に、簡便には、乱数などを用いてランダムに分類番号を付与することも可能である。そして、個数データ供給モジュールでは、このような第1の対応関係を参照しながら、各画素群の分類番号と画素群階調値とに基づいて個数データを画素群毎に決定した後、プリンタ20に供給する。

#### 【0055】

詳細には後述するが、画素群の画素群階調値は容易に求めることができる。また、画素群毎に分類番号を付与する場合も各画素群の分類番号を容易に決定して付与することが可能である。更に、記憶されている第1の対応関係を参照してやれば、分類番号および画素群階調値から個数データも容易に求めることができる。このことから、図1に例示した印刷システムでは、画素群毎の個数データを極めて迅速に生成することができ、生成した個数データを極めて迅速にプリンタ20に供給することができる。

#### 【0056】

また、プリンタ20では、供給された個数データに基づいて、次のようにして、画素群内の各画素についてのドット形成の有無を決定する。まず、画素群内の各画素について、画素群内でドットが形成される順番を示す順序値を、順序値記憶モジュールに記憶しておく。また、順序値および個数データの組合せと、該順序値を有する画素についてのドット形成の有無との対応関係を、第2の対応関係として第2対応関係記憶モジュールに記憶しておく。そして、画素群毎に個数データを受け取ると、画素群内の各画素についての順序値を取得して、個数データと順序値との組合せ毎に第2の対応関係を参照することにより、各画素についてのドット形成の有無を決定する。このように第2の対応関係を参照することによってドット形成の有無を決定してやれば、画素群内の各画素についてのドット形成の有無を、個数データに基づいて迅速に決定することが可能である。また、後述する条件が満足されていれば、ドット形成の有無を個数データに基づいて適切に決定することが可能であり、画質が低下することはない。

#### 【0057】

このように、図1に示した印刷システムでは、コンピュータ10からプリンタ20に個数データを供給しているために、たとえ画素数の多い画像であっても迅速にデータを供給することができる。また、個数データは、記憶されている第1の対応関係を参照しながら生成されるため、迅速に且つ簡便に生成することが可能である。更に、プリンタ20では、受け取った個数データを、第2の対応関係を参照しながら画素毎にドット形成の有無を表すデータに変換しているために、迅速に且つ簡便に個数データを変換することが可能である。従って、例えば画素数の多い画像であっても迅速に印刷することが可能となる。加えて、コンピュータ10などの高度な処理能力を有さない機器を用いても十分に実行可能な、簡素な印刷システムを構成することが可能となる。以下では、こうした印刷システムを例にとって、本発明の各種実施例について詳細に説明する。

#### 【0058】

B. 第1実施例 :

B-1. 装置構成 :

図2は、本実施例の画像処理装置としてのコンピュータ100の構成を示す説明図である。コンピュータ100は、CPU102を中心に、ROM104やRAM106などを

、バス116で互いに接続することによって構成された周知のコンピュータである。

#### 【0059】

コンピュータ100には、フレキシブルディスク124やコンパクトディスク126等のデータを読み込むためのディスクコントローラDDC109や、周辺機器とデータの授受を行うための周辺機器インターフェースPIF108、CRT114を駆動するためのビデオインターフェースVIF112等が接続されている。PIF108には、後述するカラープリンタ200や、ハードディスク118等が接続されている。また、デジタルカメラ120やカラースキャナ122等をPIF108に接続すれば、デジタルカメラ120やカラースキャナ122で取り込んだ画像を印刷することも可能である。また、ネットワークインターフェースカードNIC110を装着すれば、コンピュータ100を通信回線300に接続して、通信回線に接続された記憶装置310に記憶されているデータを取得することもできる。

#### 【0060】

図3は、本実施例のカラープリンタ200の概略構成を示す説明図である。カラープリンタ200はシアン、マゼンタ、イエロ、ブラックの4色インクのドットを形成可能なインクジェットプリンタである。もちろん、これら4色のインクに加えて、染料または顔料濃度の低いシアン（淡シアン）インクと、染料または顔料濃度の低いマゼンタ（淡マゼンタ）インクとを含めた合計6色のインクドットを形成可能なインクジェットプリンタを用いることもできる。尚、以下では場合によって、シアンインク、マゼンタインク、イエロインク、ブラックインク、淡シアンインク、淡マゼンタインクのそれぞれを、Cインク、Mインク、Yインク、Kインク、LCインク、LMインクと略称することがあるものとする。

#### 【0061】

カラープリンタ200は、図示するように、キャリッジ240に搭載された印字ヘッド241を駆動してインクの吐出およびドット形成を行う機構と、このキャリッジ240をキャリッジモータ230によってプラテン236の軸方向に往復動させる機構と、紙送りモータ235によって印刷用紙Pを搬送する機構と、ドットの形成やキャリッジ240の移動および印刷用紙の搬送を制御する制御回路260などから構成されている。

#### 【0062】

キャリッジ240には、Kインクを収納するインクカートリッジ242と、Cインク、Mインク、Yインクの各種インクを収納するインクカートリッジ243とが装着されている。インクカートリッジ242、243をキャリッジ240に装着すると、カートリッジ内の各インクは図示しない導入管を通じて、印字ヘッド241の下面に設けられた各色毎のインク吐出用ヘッド244ないし247に供給される。

#### 【0063】

図4は、インク吐出用ヘッド244ないし247におけるインクジェットノズルNzの配列を示す説明図である。図示するように、インク吐出用ヘッドの底面には、C、M、Y、Kの各色のインクを吐出する4組のノズル列が形成されており、1組のノズル列あたり48個のノズルNzが、一定のノズルピッチkで配列されている。

#### 【0064】

制御回路260は、CPUや、ROM、RAM、PIF（周辺機器インターフェース）等がバスで相互に接続されて構成されている。制御回路260は、キャリッジモータ230および紙送りモータ235の動作を制御することによってキャリッジ240の主走査動作および副走査動作を制御するとともに、コンピュータ100から供給される印刷データに基づいて、各ノズルから適切なタイミングでインク滴を吐出する制御を行う。こうして、制御回路260の制御の下、印刷媒体上の適切な位置に各色のインクドットを形成することによって、カラープリンタ200はカラー画像を印刷することができる。

#### 【0065】

また、インク滴を吐出するためにノズルに供給される駆動信号波形を制御してやれば、吐出されるインク滴の大きさを変更して、大きさの異なるインクドットを形成することも



できる。このようにしてインクドットの大きさを制御することができれば、印刷しようとする画像の領域に応じて異なる大きさのインクドットを使い分けてやることで、より高画質の画像を印刷することも可能となる。

#### 【0066】

尚、各色のインク吐出ヘッドからインク滴を吐出する方法には、種々の方法を適用することができる。すなわち、 piezo素子を用いてインクを吐出する方式や、インク通路に配置したヒータでインク通路内に泡（バブル）を発生させてインク滴を吐出する方法などを用いることができる。また、インクを吐出する代わりに、熱転写などの現象を利用して印刷用紙上にインクドットを形成する方式や、静電気を利用して各色のトナー粉を印刷媒体上に付着させる方式のプリンタを使用することも可能である。

#### 【0067】

以上のようなハードウェア構成を有するカラープリンタ200は、キャリッジモータ230を駆動することによって、各色のインク吐出用ヘッド244ないし247を印刷用紙Pに対して主走査方向に移動させ、また紙送りモータ235を駆動することによって、印刷用紙Pを副走査方向に移動させる。制御回路260は、キャリッジ240の主走査および副走査の動きに同期させながら、適切なタイミングでノズルを駆動してインク滴を吐出することによって、カラープリンタ200は印刷用紙上にカラー画像を印刷している。

#### 【0068】

尚、カラープリンタ200にも、制御回路260内にはCPU、RAM、ROMなどが搭載されていることから、コンピュータ100が行う処理をカラープリンタ200内で実施することも可能である。このような場合は、デジタルカメラ120などで撮影した画像の画像データをカラープリンタ200に直接供給して、制御回路260内で必要な画像処理を実施することにより、カラープリンタ200から直接画像を印刷することも可能となる。

#### 【0069】

B-2. 第1実施例の画像印刷処理の概要 :

以下では、上記のようなコンピュータ100およびカラープリンタ200が、画像を印刷するために、それぞれの内部で行われる画像処理（画像印刷処理）について説明する。ここでは、理解の便宜を図るため、初めに画像印刷処理の全体像について簡単に説明し、次に、こうした画像印刷処理が可能である原理について説明する。そして最後に、それぞれの処理の詳細な内容について説明する。

#### 【0070】

尚、以下では、画像印刷処理の前半部分はコンピュータ100で実施され、後半部分はカラープリンタ200で実施されるものとして説明するが、これらの処理をカラープリンタ200の内部で実施したり、あるいはデジタルカメラ120等の画像データを生成する機器の内部で実施することも可能である。もちろん、画像印刷処理の前半部分の処理をデジタルカメラ120などで実施して、後半部分の処理をカラープリンタ200で実施することとしても良い。すなわち、本実施例の画像印刷処理によれば、後ほど詳細に説明するように、前半部分の処理および後半部分の処理をたいへん簡素なものとして行うことができる。このため、カラープリンタ200やデジタルカメラ120などのように、コンピュータ100のように高い処理能力を有していない機器を用いた場合でも、画像印刷処理を迅速に実施することが可能であり、従って、コンピュータ100を使用せずとも十分に実用的な印刷システムを構成することができる。

#### 【0071】

図5は、第1実施例の画像印刷処理の全体的な流れを示すフローチャートである。以下では、図5を参照しながら、画像印刷処理の全体像について簡単に説明する。第1実施例の画像印刷処理を開始すると、先ず初めに、コンピュータ100が画像データの読み込みを開始する（ステップS100）。ここでは、画像データはRGBカラー画像データであるものとして説明するが、カラー画像データに限らず、モノクロ画像データについても同様に適用することができる。また、カラープリンタに限らず単色プリンタについても同様

に適用することが可能である。

#### 【0072】

カラー画像データの読み込みに続いて、色変換処理を行う（ステップS102）。色変換処理とは、R、G、Bの階調値の組合せによって表現されているRGBカラー画像データを、印刷のために使用されるインク各色についての階調値の組合せによって表現された画像データに変換する処理である。前述したように、カラープリンタ200はC、M、Y、Kの4色のインクを用いて画像を印刷している。そこで、第1実施例の色変換処理ではRGB各色によって表現された画像データを、C、M、Y、Kの各色の階調値によって表現されたデータに変換する。色変換処理は、色変換テーブル（LUT）と呼ばれる3次元の数表を参照することで行う。LUTには、RGBカラー画像データに対して、色変換によって得られるC、M、Y、K各色の階調値が予め記憶されている。ステップS102の処理では、このLUTを参照することにより、RGBカラー画像データをC、M、Y、K各色の画像データに迅速に色変換することが可能となっている。

#### 【0073】

色変換処理を終了すると、解像度変換処理を開始する（ステップS104）。解像度変換処理とは、画像データの解像度を、プリンタ200が画像を印刷する解像度（印刷解像度）に変換する処理である。画像データの解像度が印刷解像度よりも低い場合は、補間演算を行って画素間に新たな画像データを生成し、逆に画像データの解像度が印刷解像度よりも高い場合には、一定の割合でデータを間引くことによって、画像データの解像度を印刷解像度に一致させる処理を行う。

#### 【0074】

以上のようにして解像度を印刷解像度に変換したら、コンピュータ100は、個数データ生成処理を開始する（ステップS106）。個数データ生成処理の詳細な内容は後ほど詳しく説明することとして、ここでは概要のみを説明する。個数データ生成処理では、隣接する画素を所定個数ずつ画素群としてまとめることにより、1つ画像を複数の画素群に分割する。そして、それぞれの画素群の中で形成すべきドットの個数を表すデータ、すなわち個数データを画素群毎に決定する。一般に、ある画素にドットが形成されるか否かは、その画素の画像データに依存して決まるから、画素群内に形成されるドットの個数を表す個数データについても、画素群についての画像データに基づいて決定することができる。こうして各画素群について決定した個数データを、カラープリンタ200に向かって出力する。個数データ生成処理では、このようにして、各画素についての画像データに基づいて個数データを画素群毎に生成した後、カラープリンタ200に供給する処理を行う。

#### 【0075】

カラープリンタ200の制御回路260に内蔵されたCPUは、コンピュータ100から供給された個数データを受け取ると、ドット形成有無決定処理を開始する（ステップS108）。詳細な処理内容については後述するが、ドット形成有無決定処理では大まかには次のような処理を行う。上述したように、コンピュータ100から供給される個数データは、画素群に形成すべきドットの個数を表すデータであり、その個数のドットを画素群内のいずれの画素に形成するかについては未確定な状態となっている。そこで、画像を印刷するに先立って、画素群内の各画素についてドット形成の有無を予め決定しておく必要がある。ドット形成有無の決定に際しては、画素群内の各画素についてドットが形成される順番を示す順序値を記憶しておき、各画素の順序値と個数データとに基づいてドット形成の有無を決定することで、迅速に決定することが可能となっている。ドット形成有無を決定する処理の詳細については後述する。

#### 【0076】

以上のようにして、画素群内の各画素についてドット形成の有無を決定したら、決定したドット形成の有無に従って、出力媒体上にドットを形成する処理を行う（ステップS110）。すなわち、図3を用いて説明したように、キャリッジ240の主走査および副走査を繰り返しながらインク吐出用ヘッドを駆動してインク滴を吐出することにより、印刷用紙上にインクのドットを形成する。こうしてドットを形成することにより、画像データ

に対応した画像が印刷されることになる。

#### 【0077】

このように、第1実施例の画像印刷処理では、コンピュータ100からカラープリンタ200に向かって、画素群に形成すべきドット個数のデータのみを供給しており、画素群内でドットが形成される画素位置についてのデータまでは供給していない。画像を構成する画素毎にドット形成の有無を表現することに比べれば、複数の画素をまとめた画素群に形成するドットの個数は遙かに少ないデータ量で表現することができることから、このような方法を採用することで、コンピュータ100からカラープリンタ200に向かって極めて迅速にデータを供給することが可能となる。

#### 【0078】

例えば、1つの画素群に8つの画素がまとめられており、形成可能なドットは1種類であるとする。この場合、各画素はドットが形成されるか否かのいずれかの状態しか取り得ないから、画素あたり1ビットのデータ長があれば表現することができる。ここでは画素群には8つの画素が含まれているとしているから、画素群内の全画素についてドット形成の有無を表現しようとする、8ビットのデータ長が必要となる。一方、画素群内に形成されるドットの個数は、0個～8個のいずれかの9通りしか取り得ない。9通りであれば4ビットあれば表現することができるから、画素群に形成するドットの個数は4ビットのデータ長で表現可能となる。このように、画素毎にドット形成の有無を表すデータに比べて、画素群内に形成されるドット個数は遙かに少ないデータ量で表現することができるので、コンピュータ100からカラープリンタ200に向かって極めて迅速にデータを供給することが可能となるのである。

#### 【0079】

加えて、詳細には後述するが、ドットを形成する画素位置を適切に決定してやれば、ドット個数のデータのみを供給した場合でも、画質が悪化することはない。特に、所定の条件下では、画素毎にドット形成の有無を表すデータを供給した場合と全く同じ結果を得ることが可能である。

#### 【0080】

更に、後述するアルゴリズムを用いれば、個数データを生成する処理や、個数データから各画素についてのドット形成の有無を決定する処理は、極めて簡素な処理によって実現することができ、しかも極めて迅速に実行することが可能である。このため、コンピュータ100のような高度な処理能力を有する画像処理装置を用いずとも、例えばデジタルカメラ120やカラープリンタ200などの内部で実行することも可能である。こうした場合には、デジタルカメラ120で撮影した画像データを直接カラープリンタ200に供給して、高画質なカラー画像を印刷することも可能となる。

#### 【0081】

B-3. 個数データから画素位置を決定可能な原理 :

以下では、上述した方法を採用した場合、すなわち、コンピュータ100からは画素群に形成するドット個数のデータを供給し、このドット個数のデータに基づいて、画素毎にドット形成の有無をカラープリンタ200側で判断した場合でも、画質を悪化させることなく画像を印刷することが可能な原理について説明する。

#### 【0082】

説明の都合上、先ず初めに、ディザ法について説明する。ディザ法とは、画像データを画素毎にドット形成の有無を表すデータに変換するために用いられる代表的な手法である。この手法では、ディザマトリックスと呼ばれるマトリックスに閾値を設定しておき、画像データの階調値とディザマトリックスに設定されている閾値とを画素毎に比較して、画像データの階調値の方が大きい画素についてはドットを形成すると判断し、そうでない画素についてはドットを形成しないと判断する。このような判断を画像中の全画素について行えば、画像データを画素毎にドット形成の有無を表すデータに変換することができる。

#### 【0083】

図6は、ディザマトリックスの一部を概念的に例示した説明図である。図示したマトリ

ックスには、横方向（主走査方向）に128画素、縦方向（副走査方向）に64画素、合計8192個の画素に、階調値1～255の範囲から万遍なく選択された閾値がランダムに記憶されている。ここで、閾値の階調値が1～255の範囲から選択されているのは、本実施例では、画像データが階調値0～255の値を取り得る1バイトデータとしていることに加えて、画像データの階調値と閾値とが等しい場合には、その画素にはドットを形成するものと判断していることによるものである。

#### 【0084】

すなわち、ドットが形成されるのは画像データの階調値が閾値よりも大きい画素に限る（すなわち階調値と閾値が等しい画素にはドットは形成しない）とした場合、画像データの取り得る最大階調値と同じ値の閾値を有する画素には、決してドットが形成されることはない。こうしたことを避けるため、閾値の取り得る範囲は、画像データの取り得る範囲から最大階調値を除いた範囲とする。逆に、画像データの階調値と閾値が等しい画素にもドットを形成するとした場合、画像データの取り得る最小階調値と同じ値の閾値を有する画素には、常にドットが形成されてしまうことになる。こうしたことを避けるため、閾値の取り得る範囲は、画像データの取り得る範囲から最小階調値を除いた範囲とする。本実施例では、画像データの取り得る階調値が0～255であり、画像データと閾値が等しい画素にはドットを形成するとしていることから、閾値の取り得る範囲を1～255としておくのである。尚、ディザマトリックスの大きさは、図6に例示したような大きさに限られるものではなく、縦と横の画素数が同じマトリックスも含めて種々の大きさとすることができる。

#### 【0085】

図7は、ディザマトリックスを参照しながら、各画素についてのドット形成の有無を判断している様子を概念的に示した説明図である。ドット形成の有無を判断するに際しては、先ず、判断しようとする画素を選択し、この画素についての画像データの階調値と、ディザマトリックス中で対応する位置に記憶されている閾値と比較する。図7中に示した細い破線の矢印は、画像データの階調値と、ディザマトリックスに記憶されている閾値とを、画素毎に比較していることを模式的に表したものである。例えば、画像データの左上隅の画素については、画像データの階調値は97であり、ディザマトリックスの閾値は1であるから、この画素にはドットを形成すると判断する。図7中に実線で示した矢印は、この画素にはドットを形成すると判断して、判断結果をメモリに書き込んでいる様子を模式的に表したものである。一方、この画素の右隣の画素については、画像データの階調値は97、ディザマトリックスの閾値は177であり、閾値の方が大きいので、この画素についてはドットを形成しないと判断する。ディザ法では、こうしてディザマトリックスを参照しながら、画素毎にドットを形成するか否かを判断することで、画像データを画素毎にドット形成の有無を表すデータに変換する。

#### 【0086】

図8は、ディザ法を用いて画像データをドット形成の有無を表すデータに変換している様子を示した説明図である。図8（a）は、画像データの一部を拡大して示したものであり、図中の小さな矩形は画素を、そして、それぞれの矩形の中に表示された数値は画像データの階調値を表している。図示されているように、画像データは、隣接する画素間では近似する（若しくは同一の）階調値が割り当てられる傾向がある。こうした傾向は、高画質化の要請から画像データの解像度は高くなる傾向にあるが、隣接する画素間で近似若しくは同一の階調値が割り当てられる傾向は、画像データの解像度が高くなるほど顕著となっている。

#### 【0087】

図8（b）は、ディザマトリックスの対応する位置に閾値が設定されている様子を示している。図8（a）に示した画像データの階調値と、図8（b）に示したディザマトリックスの閾値とを画素毎に比較することによって、ドット形成の有無を判断する。図8（c）は、こうして画素毎にドット形成の有無を判断した結果を示しており、図中で斜線を付した画素がドットを形成すると判断された画素である。

**【0088】**

ここで、隣接する画素を所定数ずつ画素群としてまとめ、画素群内でドットを形成すると判断された画素の個数を数えることを考える。一例として、主走査方向（図8中では横方向）に4画素分、副走査方向（図8中では縦方向）に2画素分の、合計8画素ずつを画素群としてまとめるものとする。図8（d）は、こうしてまとめられたそれぞれの画素群について、ドットを形成すると判断された画素を数えることによって得られたドット個数を示している。第1実施例の画像印刷処理において、コンピュータ100からカラープリンタ200に供給されるのは、このような画素群毎の個数のデータである。個数データには、ドットを形成する画素位置に関する情報は含まれていないが、次のようにすれば、個数データからドットを形成する画素位置の情報を復元して、画素毎にドット形成の有無を表すデータを生成することができる。

**【0089】**

図9は、個数のデータから、画素毎にドット形成の有無を表すデータを生成する様子を示した説明図である。図9（a）は、図8で画素群毎に形成するドットの個数を数えて得られた値を表している。また、図9（b）は、図8で画素毎にドット形成の有無を判断するために参照したディザマトリックスを示している。前述したようにディザ法では、画像データの階調値と、ディザマトリックスの対応する画素位置に設定された閾値とを比較して、画像データの階調値の方が大きければ、その画素にはドットを形成すると判断しており、ディザマトリックスの閾値が小さくなるほどドットが形成され易くなる。このことから、ディザマトリックスはドットが形成される画素の序列を表していると考えることができる。

**【0090】**

ディザマトリックスの有するこうした性質に着目すれば、画素群内に形成されるドットの個数から、ドットが形成される画素位置を決定することができる。例えば、図9（a）に示した一番左上隅の画素群について説明すると、この画素群に形成されるドット個数は3である。また、図9（b）に示したディザマトリックスを参照すれば、この画素群内では、左上隅にある画素位置すなわち閾値「1」が設定されている画素位置が最もドットが形成され易い画素であると言える。従って、この画素群内で3つ形成されるドットの中の1つは、左上隅の画素に形成されているものと考えることができる。同様にして、残りの2つのドットは、この画素群内で2番目にドットが形成され易い画素（すなわち図9（b）のディザマトリックスで閾値「42」が設定されている画素）と、3番目にドットが形成され易い画素（すなわち閾値「58」が設定されている画素）とに形成されるものと考えることができる。

**【0091】**

もちろん、ドット形成の有無は、ディザマトリックスに設定された閾値だけでなく、画像データの階調値によっても影響されるから、画像データの階調値が極端に大きければ、より小さな閾値が設定されている画素よりも先にドットが形成されることも起こり得る。しかし、前述したように画像データには、隣接する画素には近似する（若しくは同一の）階調値が割り当てられる傾向があるから、ほとんどの場合はドットが形成され易い画素（すなわちディザマトリックスに設定された閾値の小さな画素）からドットが形成されると考えることができる。

**【0092】**

図9（a）に示した他の画素群についても、同様にして、ドット個数とディザマトリックスの閾値とに基づいて、ドットを形成する画素位置を決定することができる。例えば、図9（a）の上述した画素群の下にある画素群（左端上から2番目の画素群）については、ドット個数は3個であるから、図9（b）のディザマトリックスを参照すれば、これら3つのドットは、閾値「22」が設定された画素と、閾値「33」が設定された画素と、閾値「91」が設定された画素とに、それぞれ形成されると考えることができる。

**【0093】**

図9（a）に示した4つの画素群について、このようにして個数データからドットを形

成する画素位置を決定すると、図9(c)に示した結果を得ることができる。図9(c)中で、斜線を付して示した画素はドットを形成すると判断された画素である。図9(c)と図8(c)とを比較すれば明かなように、個数データから決定した画素位置は、画素毎に決定した画素位置と一致している。このことは、ディザマトリックスを参照して画素毎にドット形成の有無を判断し、画素群内に形成されるドットの個数のみを記憶しておけば、画素位置までは記憶していなくても、ディザマトリックスとドット個数とから、ドットが形成される画素位置を復元可能なことを示している。このことから、コンピュータ100から画素群毎の個数データを供給し、カラープリンタ200側で個数データからドットを形成する画素位置を決定した場合でも、画素位置を適切に決定して、画質を悪化させることなく画像を印刷することが可能となるのである。

#### 【0094】

また、個数データからドットを形成する画素位置を適切に決定するためには、画像データの階調値が画素群内で大きく異ならなければ良い。前述したように、画像データは隣接する画素間では近似した階調値を有する特性があるから、こうした条件はほとんどの場合に成立しており、従って、個数データのみをカラープリンタ200に供給した場合でも、画質を悪化させることなく画像を印刷することができるのである。

#### 【0095】

特に、次の2つの条件が満足される場合には、画像データの階調値とディザマトリックスの閾値とを比較してドット形成の有無を画素毎に判断した結果と、完全に同じ画素位置にドットが形成されることが保証される。まず1つ目の条件は、画素群内で各画素の階調値が同一の値を有することであり、2つ目の条件は、コンピュータ100側で画素毎にドット形成の有無を判断する際に参照したディザマトリックスと、カラープリンタ200側で個数データから画素位置を決定するために参照するディザマトリックスとが、同一のマトリックスであることである。

#### 【0096】

尚、図7で説明したディザ法においては、ディザマトリックスに設定された閾値と画像データとの階調値とを比較して、いずれの値が大きいかによってドット形成の有無を判断している。これに対して、個数データから画素群内でドットが形成される画素位置を決定する場合には、図9を用いて説明したように、ドットが形成される画素位置を、ディザマトリックスに設定された閾値の小さな画素から順番に決定している。すなわち、画素位置を決定するためには、閾値の値まで必要なわけではなく、画素群内でドットが形成され易い順序（換言すれば、ドットが形成される順番）が分かっているれば良い。このことから、図9(b)に示すディザマトリックスの代わりに、図9(d)に示すような画素群内の各画素について、ドットが形成される順番を示す値（順序値）が設定されたマトリックス（本明細書中では、このようなマトリックスを順序値マトリックスと呼ぶものとする）を記憶しておき、画素群毎に順序値マトリックスを参照しながら、個数データから画素位置を決定することも可能である。

#### 【0097】

また、このように、ディザマトリックスに基づいてドット形成の有無を決定する場合、ディザ法を適用してドット形成の有無を決定した時とほぼ同等のドット分布を得ることができる。このことから、ディザマトリックスの特性を適切に設計しておくことで、ドットの分布を制御することが可能である。例えば、いわゆるブルーノイズマスク特性を有するマトリックスや、あるいはグリーンノイズマスク特性を有するマトリックスを使用すれば、画像データを画素群単位で処理しているにも関わらず、これらディザマトリックスの特性に依存したドット分布の画像を得ることができる。

#### 【0098】

図10は、ブルーノイズマスク特性を有するディザマトリックス、およびグリーンノイズマスク特性を有するディザマトリックスについて、設定されている閾値の空間周波数特性を概念的に例示した説明図である。図10では、表示の都合から、横軸には空間周波数の代わりに周期を取って表示している。言うまでもなく、周期が短くなるほど、空間周波



数は高くなる。また、図10の縦軸は、それぞれの周期での空間周波数成分を示している。尚、図示されている周波数成分は、ある程度変化が滑らかとなるように平滑化された状態で示されている。

#### 【0099】

図中の実線は、ブルーノイズマスクの空間周波数成分を概念的に示している。図示されているように、ブルーノイズマスクは、1周期の長さが2画素以下の高い周波数領域に最も大きな周波数成分を有している。ブルーノイズマスクの閾値は、このような空間周波数特性を有するように設定されていることから、ブルーノイズマスクに基づいてドット形成の有無を判断した場合には、ドットが互いに離れた状態で形成される傾向にある。また、図中の破線は、グリーンノイズマスクの空間周波数成分を概念的に示している。図示されているように、グリーンノイズマスクは、1周期の長さが2画素から十数画素の中間周波数領域に最も大きな周波数成分を有している。グリーンノイズマスクの閾値は、このような空間周波数特性を有するように設定されていることから、グリーンノイズマスクに基づいてドット形成の有無を判断した場合には、数ドット単位で隣接してドットが形成されながら、全体としてはドットの固まりが分散した状態で形成される傾向にある。

#### 【0100】

従って、このようなブルーノイズマスク特性、あるいはグリーンノイズマスク特性を有するディザマトリックスに基づいて、画素群の個数データを決定したり、あるいは、画素位置を決定してやれば、画素群単位で処理しているにも関わらず、ブルーノイズマスク特性あるいはグリーンノイズマスク特性を反映した分布で、ドットを形成することが可能となる。

#### 【0101】

また、以上の説明では、図9に示したように、ディザマトリックスに基づいて生成されて、複数種類の順序値マトリックスを予め記憶しておき、画素群の個数データを受け取ると、その画素群に対応した順序値マトリックスを用いて、各画素についてのドット形成の有無を決定するものとして説明した。しかし、より簡便には次のようにしてドット形成の有無を決定しても良い。すなわち、予め複数の順序値マトリックスを記憶しておき、個数データを受け取ると、画素群毎にランダムに選択した1の順序値マトリックスを用いて、各画素についてのドット形成の有無を決定しても良い。更に、より簡便には、順序値マトリックスを1組だけ記憶しておき、このマトリックスを用いて各画素についてのドット形成の有無を決定することも可能である。

#### 【0102】

##### B-4. 第1実施例の個数データ生成処理：

以下では、図5に示した第1実施例の画像印刷処理において、画像データから個数データを生成する処理（ステップS106）について説明する。図11は、第1実施例の個数データ生成処理の流れを示すフローチャートである。ここでは、個数データ生成処理はコンピュータ100で実施されるものとして説明するが、後述するように、個数データ生成処理は極めて簡素な処理とすることができるので、カラープリンタ200あるいはデジタルカメラ120などの内部で実施することも可能である。以下、フローチャートに従って説明する。

#### 【0103】

第1実施例の個数データ生成処理を開始すると、まず初めに、互いに隣接する所定個数の画素をまとめて画素群を生成する（ステップS200）。ここでは、主走査方向に4画素分、副走査方向に2画素分の合計8つの画素を画素群にまとめるものとする。尚、画素群としてまとめる画素は、このように矩形状に縦横の位置が揃った画素である必要はなく、互いに隣接し且つ所定の位置関係にある画素であれば、どのような画素を画素群としてまとめても良い。

#### 【0104】

次いで、画素群の分類番号と、画素群階調値とを決定する（ステップS202）。画素群の分類番号は、後述する方法を用いれば、極めて簡便に決定することができる。また、

画素群階調値についても、次のようにして簡単に決定することができる。例えば、画素群内の各画素に割り当てられた階調値の平均値を求めて画素群階調値とすることができるし、あるいは画素群の中で最も多くの画素に割り当てられた階調値、更には、画素群内で特定の位置にある画素の階調値を画素群階調値とすることもできる。

#### 【0105】

こうして、画素群の分類番号と画素群階調値とを決定したら、後述する第1の変換テーブルを参照することによって個数データを決定する（ステップS204）。詳細には後述するが、第1の変換テーブルには、画素群の分類番号と画素群階調値との組合せに対応付けて、適切な個数データが予め記憶されている。従って、画素群の分類番号と画素群階調値とが決定されれば、第1の変換テーブルを参照することで直ちに個数データを求めることが可能である。この点についても、後ほど詳しく説明する。

#### 【0106】

以上のようにして、1つの画素群について個数データが得られたら、画像データの全画素について処理を終了したか否かを判断する（ステップS206）。そして、未処理の画素が残っている場合は（ステップS206：no）、ステップS200に戻って新たな画素群を生成し、続く一連の処理を繰り返す。こうした操作を繰り返し、全画素についての処理が終了したと判断されたら（ステップS206：yes）、各画素群について得られた個数データをカラープリンタ200に出力して（ステップS208）、図11に示す第1実施例の個数データ生成処理を終了する。

#### 【0107】

### B-5. 分類番号の決定方法：

ここで、上述した第1実施例の個数データ生成処理中で、画素群の分類番号を決定する方法について説明する。以下では、先ず初めに、画素群の分類番号を付与する考え方を説明した後、分類番号を付与する具体的な方法について説明する。

#### 【0108】

図12は、画素群毎の分類番号を決定するための考え方を示した説明図である。図12(a)は、画像の一番左上隅の箇所において、横方向に4画素、縦方向に2画素の合計8画素をまとめることによって画素群を1つ生成した様子を概念的に示したものである。

#### 【0109】

前述したように、ディザ法では画素に割り当てられた画像データの階調値と、ディザマトリックスの対応する位置に設定されている閾値とを比較して、画素毎にドット形成の有無を判断している。一方、本実施例では、隣接する所定数の画素を画素群としてまとめているから、ディザマトリックスに設定されている閾値についても、画素群に対応する所定数ずつまとめてブロックを生成することにする。図12(b)は、図6に示したディザマトリックスに設定されている閾値を、横方向に4つ、縦方向に2つずつまとめて複数のブロックを生成した様子を示している。図6に示したディザマトリックスは、横方向（主走査方向）に128画素分、縦方向（副走査方向）に64画素分の合計8192画素分の閾値が設定されているから、これら閾値を横方向に4つ、縦方向に2つずつブロックにまとめれば、ディザマトリックスは縦横それぞれ32個ずつ、合計1024個のブロックに分割されることになる。

#### 【0110】

今、図12(b)に示すように、これらブロックに1番～1024番までの通し番号を付しておく。そして、画像データにディザマトリックスを適用した時に、各画素群の位置に適用されるブロックの通し番号によって、画素群を分類してやる。例えば、図12(c)に示したように、画像の一番左上隅にある画素群には、図12(b)中の通し番号1番のブロックが適用されるから、この画素群は分類番号1番の画素群に分類するのである。

#### 【0111】

以上が、画素群を分類する際の基本的な考え方である。図11のステップS202では、このように、画像データにディザマトリックスを適用したときに、画素群に適用されるブロックの通し番号によって各画素群を分類し、対応する分類番号を決定して画素群に付



与する処理を行う。

#### 【0112】

次に、画素群の分類番号を決定するための具体的な方法について説明する。図13は、画素群の分類番号を決定する方法を示した説明図である。図13(a)は、画像中で生成された1つの画素群を表している。ここでは、この画素群に着目して分類番号を決定する方法について説明する。尚、以下では、分類番号を決定するために着目している画素群を、着目画素群と呼ぶことにする。

#### 【0113】

今、画像の一番左上隅にある画素を原点に取って、原点からの主走査方向および副走査方向への画素数によって画素位置を表すものとする。また、画素群の位置は、画素群の左上隅にある画素の画素位置によって表すものとする。図13(a)では、着目画素群の位置を示す画素に黒丸を付して表示している。この画素の画素位置が(X, Y)であったとする。すると、各画素群の大きさは、主走査方向に4画素、副走査方向に2画素としているから、

$$X = 4n + 1, \quad Y = 2m + 1$$

となるようなn, m(ここで、n, mは0以上の正整数)が存在する。換言すれば、着目画素群の左側にはn個の画素群が並んでおり、着目画素群の上側にはm個の画素群が並んでいることになる。

#### 【0114】

ここで、前述したように画素群は、画像データにディザマトリックスを適用したときに、着目画素群に適用されるブロックの通し番号に基づいて分類することとしているから(図12参照のこと)、ディザマトリックスを移動させながら画像データに適用する方法によって、同じ画素群でも異なった分類番号に分類されることになる。実際には、ディザマトリックスを移動させながら画像データに適用する方法はどのような方法でも構わないが、ここでは説明の便宜から、最も単純な方法すなわちディザマトリックスを横方向に移動させるものとして説明する。図13(b)には、ディザマトリックスを横方向に少しずつ移動させながら、繰り返し画像データに適用している様子が概念的に示されている。

#### 【0115】

図13(c)は、図13(b)に示すようにディザマトリックスを繰り返して用いながら、図13(a)に示した着目画素群にディザマトリックスを適用している様子を概念的に表している。このようにディザマトリックスを移動させていくと、ディザマトリックス中のいずれかのブロックが、着目画素群に適用されることになる。ここでは、着目画素群には、ディザマトリックス中でM行N列目のブロックが適用されたものとする。すると、図13(a)に示したように着目画素群の左側にはn個の画素群があり、上側にはm個の画素群があるから、Nとn、およびMとmとの間には、それぞれ

$$\begin{aligned} N &= n - \text{int}(n/32) \times 32 + 1 \\ M &= m - \text{int}(m/32) \times 32 + 1 \end{aligned}$$

の関係が成り立っている。ここで、intは、小数点以下を切り捨てて整数化することを表す演算子である。すなわち、 $\text{int}(n/32)$ は、 $n/32$ の計算結果に対して小数点以下の数値を切り捨てることによって得られた整数値を表している。このように、着目画素群の位置が分かれば、図13(d)に表示した上述の関係式から数値MおよびNを求めて、ディザマトリックス中でM行N列目にあるブロックのブロック番号を、その着目画素群の分類番号としてやればよい。もっとも実際には、M, Nの値は、図13(d)に示すような計算を実行せずとも、極めて簡便に求めることができる。以下、この点について説明する。

#### 【0116】

図14は、着目画素群の分類番号を決定する方法を具体的に示した説明図である。着目画素群の位置を(X, Y)として、X, Yが10ビットで表現されているものとする。図14(a)は、数値Xを表す10ビットの2進数データを概念的に示している。図では、各ビットを識別するために、最上位ビットから最下位ビットに向かって1番から10番ま

での通し番号を付して表示している。

#### 【0117】

図13を用いて前述したように、着目画素群の左側にある画素群の個数 $n$ は、数値 $X$ から1を減算して4で除算すれば得ることができる。ここで、4での除算は、2ビット分だけ右方向にシフトさせることで実施することができるから、数値 $X$ から1を減算して、得られた2進数データを右方向に2ビット分だけビットシフトさせればよい。更に、数値 $X$ は任意の値を取るのではなく、 $4n+1$ の形式で表現可能な数値しか取り得ないから、1を減算せずに、単に2進数データを右方向に2ビット分だけビットシフトさせるだけで、画素群の個数 $n$ を得ることができる。図14(b)は、こうして数値 $X$ をビットシフトして得られた個数 $n$ の2進数データを概念的に表している。

#### 【0118】

次いで、 $\text{int}(n/32)$ を算出する。すなわち、個数 $n$ を32で除算して、小数点以下の数値を切り捨てる操作を行う。32による除算は、2進数データを右方向に5ビット分だけビットシフトさせることで実行可能であり、また、データを整数形式で扱っていれば、小数点以下の数値は自動的に切り捨てられてしまう。結局、 $\text{int}(n/32)$ の2進数データは、個数 $n$ の2進数データを、単に右方向に5ビット分だけビットシフトさせることで得ることができる。図14(c)は、個数 $n$ をビットシフトして得られた $\text{int}(n/32)$ の2進数データを概念的に表している。

#### 【0119】

こうして得られた $\text{int}(n/32)$ に32を乗算する。32による乗算は、2進数データを5ビット分だけ左方向にビットシフトすることで実施することができる。図14(d)は、個数 $n$ をビットシフトして得られた $\text{int}(n/32) \times 32$ の2進数データを概念的に表している。

#### 【0120】

次いで、個数 $n$ から $\text{int}(n/32) \times 32$ を減算すれば、前述の数値 $N$ を得ることができる。個数 $n$ の2進数データ(図14(b)参照)と $\text{int}(n/32) \times 32$ の2進数データ(図14(d)参照)とを比較すれば明らかなように、これら2進数データは、上位の5ビットは共通しており、減算する側の数値の下位5ビットは全て「0」となっている。従って、減算される側の数値(個数 $n$ )の下位5ビットをそのまま抜き出せば、求める数値 $M$ を得ることができる。すなわち、図14(b)に示した2進数データと、図14(f)に示すようなマスクデータとの論理積を求めるだけで、極めて簡便に数値 $N$ を得ることが可能である。あるいは、図14(a)に示した着目画素群の位置を示す数値 $X$ の2進数データと、図14(g)のようなマスクデータとの論理積を求めることにより、4番目～8番目のビットデータを直接抜き出すことによっても、数値 $N$ を得ることができる。

#### 【0121】

図14では、着目画素群の位置を示す座標値( $X$ ,  $Y$ )の数値 $X$ から、ディザマトリックス中でのブロック位置を示す数値 $N$ を求める場合について説明したが、全く同様にして、ブロック位置を示す数値 $M$ も数値 $Y$ から求めることができる。結局、着目画素群の位置が分かれば、2進数データから特定のビット位置のデータを抜き出すだけで、着目画素群がディザマトリックス中で何行何列目のブロックに対応するかを知ることができ、このブロックの通し番号によって、着目画素群の分類番号を迅速に決定することが可能なのである。

#### 【0122】

図11を用いて前述したように、第1実施例の個数データ生成処理では、こうして得られた画素群の分類番号と画素群階調値とから、第1の変換テーブルを参照することによって個数データを取得する(図11のステップS204参照)。以下では、個数データを得るために参照される第1の変換テーブルについて説明する。

#### 【0123】

図15は、画素群の分類番号と画素群階調値とから個数データを取得するために参照さ

れる第1の変換テーブルを概念的に示した説明図である。図示されているように、第1の変換テーブルには、画素群の分類番号と画素群階調値との組合せに対応付けて、適切な個数データが予め記憶されている。図12を用いて前述したように、ここでは画素群は分類番号1番～1024番のいずれかに分類され、また、画素群階調値は0～255のいずれかの階調値を取り得るとしているから、分類番号と画素群階調値との組合せは、 $1024 \times 256 = 262144$ の組合せが存在する。第1の変換テーブルには、この全ての組合せに対して個数データが設定されており、この個数データは次のようにして決定されている。

#### 【0124】

図16は、画素群の分類番号と画素群階調値との組合せに応じて、適切な個数データを決定する様子を概念的に示した説明図である。一例として、画素群の分類番号が1番であるとする。分類番号1番の画素群には、ディザマトリックス中で通し番号1番のブロックが適用される。図16(a)には、通し番号1番のブロックに設定されている閾値が示されている。

#### 【0125】

今、画素群階調値が0であるとする。この場合は、画素群中の全画素が階調値0の画像データを有するものとする。そして、各画素の階調値（すなわち「0」）と図16(a)に示した閾値とを比較して、階調値の方が大きい（若しくは同じ）画素については、ドットを形成するものと判断する。画素群の全画素についてこうした判断を行った後、ドットの個数を数えて、得られた値を個数データとする。図16(a)に示したいずれの閾値も、階調値0よりは大きいから、ドットを形成すると判断される画素は存在しない。そこで、分類番号が1番で画素群階調値が0の組合せについては、個数データ0を設定する。図16(b)は、画素群階調値が0の場合の個数データを決定している様子を概念的に表している。上述したように、この場合は、画素群内のいずれの画素にもドットは形成されず、従って、個数データは0となる。

#### 【0126】

図16(c)は、画素群階調値が1の場合に個数データを決定する様子を概念的に示している。この場合は、画素群内の全画素が階調値1の画像データを有するものとして、各画素の階調値を図16(a)に示した閾値と比較する。その結果、画素群内で左上隅にある画素では、画像データの階調値と閾値とが等しくなってドットを形成すると判断され、他の画素についてはドットを形成しないと判断される。図16(c)に表示された斜線の付された丸印は、その画素にドットを形成すると判断されたことを表している。この結果、分類番号が1番で画素群階調値が1の組合せについては、個数データ1が設定される。

#### 【0127】

こうした操作を、0～255までの全ての画素群階調値について行うことにより、個数データを決定していく。例えば、画素群階調値が2の場合は、図16(d)に示したように、個数データは1となり、画素群階調値が100の場合は、図16(e)に示すように個数データは3となる。図16(f)および図16(g)には、画素群階調値が200の場合および画素群階調値が255の場合に、それぞれの個数データを決定する様子が概念的に示されている。図15中で、分類番号1に該当する行（表中に示された横方向の欄）の部分に、各画素群階調値に対応付けて設定されている個数データは、このようにして決定された個数データである。こうした操作を、1番～1024番までの全ての分類番号について行えば、最終的に、全ての分類番号と全ての画素群階調値とのあらゆる組合せに対応する個数データを決定することができる。図15に示した第1の変換テーブルには、分類番号と画素群階調値との組合せに応じて、対応する個数データが予め設定されている。

#### 【0128】

B-6. 第1実施例のドット形成有無決定処理 :

次に、前述した第1実施例の画像印刷処理において、画素群内の各画素についてのドット形成の有無を、個数データに基づいて決定する処理（図5のステップS108）について説明する。図17は、第1実施例のドット形成有無決定処理の流れを示すフローチャー

トである。かかる処理は、カラープリンタ 200 の制御回路 260 に内蔵された CPU によって実行される処理である。また、図 18 は、第 1 実施例のドット形成有無決定処理において、各画素についてのドット形成の有無が決定される様子を概念的に示した説明図である。以下では、図 18 を参照しながら、図 17 に示すフローチャートに従って、第 1 実施例のドット形成有無決定処理について説明する。

#### 【0129】

ドット形成有無決定処理を開始すると、先ず初めに、画素群を 1 つ選択し（ステップ S300）、その画素群の個数データを取得する（ステップ S302）。ここでは、図 18（a）に示すような個数データが供給されたものとする。

#### 【0130】

次いで、選択した画素群に含まれる各画素の中から対象とする画素を 1 つ選択し（ステップ S304）、該画素群内で対象画素にドットが形成される順番を示す値（順序値）を取得する（ステップ S306）。対象画素の順序値は、図 18（b）に示すような、予め設定されている順序値マトリックスを参照することで容易に取得することができる。図 18（b）に例示した順序値マトリックスには、画素群を構成する各画素の画素位置について、順序値が予め設定されている。例えば、画素群内で一番左上隅にある画素については、順序値「1」が設定されており、その画素の右隣の画素については、順序値「6」が設定されている。ステップ S306 では、このような順序値マトリックスを参照して、対象画素の位置に設定されている順序値を取得する。尚、説明が煩雑となることを避けるために、ここでは、順序値マトリックスは 1 組のみが記憶されており、常に同じ順序値マトリックスを用いて順序値を得るものとして説明するが、順序値マトリックスを複数組記憶しておき、画素群毎に順序値マトリックスを切り換えながら、対象画素の順序値を取得することとしても良い。

#### 【0131】

こうして順序値を取得したら、第 2 の変換テーブルを参照することによって、対象画素についてのドット形成の有無を決定する（ステップ S308）。図 19 は、対象画素についてのドット形成の有無を決定するために参照される第 2 の変換テーブルを概念的に示した説明図である。前述した第 1 の変換テーブルには、分類番号と画素群階調値との組合せに対応付けて個数データが記憶されているのに対して（図 15 参照）、第 2 の変換テーブルには、図 19 に示されているように、順序値と個数データとの組合せに対応付けて、ドット形成の有無が設定されている。ここでは、画素群は 8 つの画素から構成されているから、順序値は 1～8 までの値を取り、個数データは 0～8 までの値を取るとしているから、第 2 の変換テーブルには、これらを組合せた 72 通りの組合せに対応付けて、ドット形成の有無を示す値が設定されている。図 19 に示した例では、ドットを形成する組合せに「1」が、ドットを形成しない組合せには「0」が設定されている。

#### 【0132】

一例として、図 18 に示した画素群内で一番左上隅にある画素について説明すると、図 18（b）に示すように順序値は「1」であり、個数データは図 18（a）に示したように「3」である。図 19 に例示した第 2 の変換テーブルを参照すると、順序値「1」、個数データ「3」の組合せに対して設定されている値は「1」、すなわち、この画素についてはドットを形成するものと決定することができる。このように、図 17 のステップ S308 では第 2 の変換テーブルを参照することで、画素群についての個数データと対象画素の順序値とから、対象画素についてのドット形成の有無を直ちに決定するのである。

#### 【0133】

こうして、対象画素として選択した 1 つの画素についてドット形成の有無を決定したら、選択した画素群内の全画素について、ドット形成の有無を決定したか否かを判断する（ステップ S310）。そして、画素群内で未だドット形成の有無を決定していない画素が残っている場合は（ステップ S310：no）、ステップ S304 に戻って、画素群の中から新たな画素を対象画素として選択し、続く一連の処理を行う。

#### 【0134】

こうした処理を繰り返し、画素群内の全画素についてドット形成の有無を決定したと判断されたら（ステップ S 3 1 0 : y e s）、今度は、個数データの供給された全画素群について処理を終了したか否かを判断する（ステップ S 3 1 2）。そして、未処理の画素群が残っていれば（ステップ S 3 1 2 : n o）、ステップ S 3 0 0 に戻って新たな画素群を選択し、続く一連の処理を行う。こうした処理を繰り返すことにより、コンピュータから供給された個数データは、画素毎にドット形成の有無を示すデータに変換されていく。そして、全ての画素群について処理が終了したら（ステップ S 3 0 6 : y e s）、図 1 7 に示したドット形成有無決定処理を終了して、図 5 の画像印刷処理に復帰する。

#### 【0135】

以上、第 1 実施例の画像印刷処理中で行われる個数データ生成処理（図 5 のステップ S 1 0 6）、およびドット形成有無決定処理（図 5 のステップ S 1 0 8）の内容について詳しく説明した。上述した個数データ生成処理では、所定数の画素をまとめて画素群を生成し、その画素群について分類番号と画素群階調値を決定した後、個数データを生成する。画素群の分類番号および画素群階調値は上述したように極めて容易に求めることができる。そして、分類番号および画素群階調値が分かれば、前述した第 1 の変換テーブルを参照することによって、極めて容易に個数データを生成することができる。こうして生成された個数データは、画素毎にドット形成の有無を表すデータに比べて、データ量が遙かに小さくなっているため、コンピュータ 1 0 0 からカラープリンタ 2 0 0 に向かって極めて迅速にデータを出力することができる。すなわち、上述した個数データ生成処理では、個数データの生成および出力を高速に実行することが可能となる。

#### 【0136】

また、上述したドット形成有無決定処理では、こうしてコンピュータ 1 0 0 から迅速に供給された個数データを受け取ると、先ず、順序値マトリックスを参照することによって対象画素の順序値を取得する。次いで、得られた順序値と個数データを用いて第 2 の変換テーブルを参照することによって、画素群内の各画素についてのドット形成の有無を決定する。こうすれば、順序値マトリックスと第 2 の変換テーブルを参照するだけで、ドット形成の有無を迅速に決定することが可能である。

#### 【0137】

加えて、個数データを生成する処理は、単にテーブルを参照する処理に過ぎず、第 1 の変換テーブルを参照するために使用する分類番号や画素群階調値も、極めて簡便な処理で求めることができる。同様に、個数データからドット形成の有無を決定する処理も、単にテーブルを参照する処理に過ぎない。このため、いずれの処理についても、コンピュータ 1 0 0 のような高いデータ処理能力を備えていない機器を用いた場合でも、十分に実用的な速度で処理することができる。

#### 【0138】

更に、処理の大部分は、マトリックスあるいはテーブルに記憶されているデータを単に参照するという極めて単純なものであるため、CPU を用いてソフトウェア的に実行するのではなく、専用の論理回路を組み込んだ IC チップを用いてハードウェア的に実行することも容易である。処理をハードウェア的に実行すれば、より一層高速に実行することが可能となり、その分だけ画像を迅速に印刷することも可能となる。

#### 【0139】

また、近年のコンピュータでは処理の高速化を図るために、いわゆるパイプライン処理と呼ばれる技術が活用されているが、その一方で、条件分岐を含む処理に対しては、パイプライン処理の技術を適用しても、さほど処理を高速化することができず、返って処理速度が低下する場合もあることが知られている。上述した個数データ生成処理およびドット形成有無決定処理は、記憶されているデータを参照する処理がほとんどであり、条件分岐が含まれていないので、こうした観点からも、高速処理に適した処理であると言える。

#### 【0140】

このように、上述した第 1 実施例の個数データ生成処理およびドット形成有無決定処理

は、迅速な処理を可能とする種々の要素を備えているため、どのような条件においても、画像データを迅速に処理することができ、延いては画像を迅速に印刷することが可能となる。

#### 【0141】

B-7. 第1の変形例：

上述した第1実施例の画像印刷処理では、画像データを印刷解像度に変換した後、所定数の画素をまとめて画素群を形成して個数データを生成した。しかし、画像データを印刷解像度よりも低解像度のデータに一旦変換し、明示的には画素群を形成することなく個数データを生成することとしても良い。以下では、こうした第1実施例の第1の変形例について説明する。

#### 【0142】

図20は、第1の変形例の画像印刷処理の流れを示したフローチャートである。第1の変形例の画像印刷処理は、図5に示した第1実施例の画像印刷処理に対して、画像データの解像度を印刷解像度に変換するのではなく、印刷解像度よりも低解像度のデータに変換する点と、個数データ生成処理において明示的には画素群を形成していない点が大きく異なっているが、他の点についてはほぼ同様である。以下では、第1実施例の画像印刷処理との相違点を中心として、第1の変形例の画像印刷処理について説明する。

#### 【0143】

第1の変形例の画像印刷処理においても、第1実施例の画像印刷処理と同様に、処理を開始すると先ず初めに画像データを読み込んで（ステップS400）、色変換処理を行う（ステップS402）。

#### 【0144】

次いで、第1の変形例の画像印刷処理では、画像データに解像度調整処理を施して、印刷解像度よりも低解像度の画像データに一旦変換する（ステップS404）。図21は、解像度調整処理で行われる処理について示した説明図である。図21（a）は、色変換後の画像データを概念的に表しており、図21（b）は、解像度調整処理によって生成される画像データを概念的に表している。また、図21（c）は、印刷解像度の画像データを示している。図21（b）と図21（c）とを比較すれば明らかなように、解像度調整処理によって生成される画像データは、印刷解像度よりも低解像度のデータである。より詳しくは、解像度調整処理によって生成される画像データの解像度は、主走査方向には印刷解像度の $1/4$ 、副走査方向には印刷解像度の $1/2$ の解像度となっている。換言すれば、解像度調整処理によって生成される画素は、図21（c）に示した印刷解像度の画素に対して、主走査方向には4倍、副走査方向には2倍の大きさを有する大きな画素に変換される。そして、続いて行われる個数データ生成処理では、図21（b）に示す大きな画素が、あたかも図21（c）に示すように印刷解像度の画素を所定数ずつまとめて生成した画素群であるかのように扱って、個数データを生成する。すなわち、変形例の解像度調整処理では、解像度変換後の1画素の大きさが、印刷解像度の画素をまとめて生成した画素群の大きさと一致するように、画像データの解像度を変換する処理を行うのである。

#### 【0145】

変形例の画像印刷処理では、こうして色変換後の画像データの解像度を調整した後、個数データ生成処理を開始する（図20のステップS406）。図22は、第1の変形例の画像印刷処理で行われる個数データ生成処理の流れを示すフローチャートである。かかる処理を開始すると、先ず初めに、処理対象とする画素を1つ選択する（ステップS450）。ここで選択する画素は、図21（b）に示すような、印刷解像度の画素よりも大きな画素である。しかし、この画素の大きさは、前述した第1実施例において印刷解像度の画素をまとめて生成した画素群の大きさと一致している。そこで、選択した画素をあたかも第1実施例における画素群のように扱って、この画素についての分類番号を決定する（ステップS452）。分類番号は、図12ないし図14における画素群を画素と読み替えてやれば、第1実施例における方法に準じて決定することができる。

#### 【0146】



次いで、図 15 に示した第 1 の変換テーブルを参照することにより、選択した画素についての個数データを取得する（ステップ S 4 5 4）。第 1 の変換テーブルを参照する際に用いる画素群階調値には、選択した画素に割り当てられている画像データの階調値をそのまま使用することができる。

#### 【0147】

こうして、処理対象として選択した画素について個数データが取得されたら、全画素について処理を終了したか否かを判断する（ステップ S 4 5 6）。未処理の画素が残っている場合は（ステップ S 4 5 6：no）、ステップ S 4 5 0 に戻って新たな処理対象の画素を選択した後、続く一連の処理を行う。こうした操作を繰り返し、全画素について処理を終了したと判断されたら（ステップ S 4 5 6：yes）、各画素について得られた個数データをカラープリンタ 200 に出力して（ステップ S 4 5 8）、図 22 に示す第 1 の変形例の個数データ生成処理を終了する。

#### 【0148】

以上のような個数データ生成処理に続いて、ドット形成有無決定処理を行う（ステップ S 4 0 8）。ドット形成有無決定処理については、第 1 の変形例の画像印刷処理においても前述した第 1 実施例の画像印刷処理と同様である。すなわち、コンピュータ 100 から供給された個数データを受け取って、順序値マトリックスを参照することにより画素群内の各画素についての順序値を求める。そして、得られた順序値および個数データから第 2 の変換テーブルを参照することにより、各画素についてのドット形成の有無を決定する。

#### 【0149】

次いで、こうして決定された画素位置にドットを形成していく（ステップ S 4 1 0）。その結果、印刷用紙上に適切な密度でドットが形成されて、画像データに対応する画像が印刷されることになる。

#### 【0150】

上述した変形例の画像印刷処理では、画像データを印刷解像度よりも低い解像度のデータとしたまま、個数データを生成することができる。解像度が低くなれば画像データのデータ量は小さくなるので、その分だけデータを迅速に取り扱うことが可能になるとともに、処理に際して一時的に必要なメモリ量も低減させることも可能となる。また、画素群を生成する処理や、生成した画素群についての画素群階調値を算出する処理も不要となるので、処理の簡素化と迅速化とを同時に実現することが可能となる。

#### 【0151】

尚、高画質な画像を印刷するためには、高い解像度で印刷することが効果的であることは言うまでも無いが、必ずしも、印刷解像度を高くすることに合わせて画像データの解像度も高くする必要があるわけではない。低解像度の画像データを受け取って、単に大きな画素を小さな画素に分割し、見かけ上の解像度を高くするだけでも、印刷画質を改善することが可能である。例えば、図 21（b）に示すような低解像度の画像データを受け取って、各画素を複数の画素に分割し、図 21（c）に示すような高解像度の画像データに変換する。こうして得られた画像データは見かけ上の解像度は高くなっているものの、解像度に見合うだけ滑らかな階調変化が可能になっているわけではなく、階調変化を滑らかに表現する観点から見れば低解像度の画像データと何ら変わるところはない。しかし、次のような理由から、見かけ上の解像度を高くするだけでも印刷画質を向上させることが可能であるため、こうした処理も現在では比較的頻繁に行われるようになっている。

#### 【0152】

以下では、見かけ上の解像度を向上させるだけでも印刷画質が改善される理由について簡単に説明する。画像データは、一般的に、個々の画素について多階調を表現することが可能である。例えば、画像データが 1 バイトデータである場合には、画素あたり 256 階調を表現することができる。これに対して、ドットを形成して画像を印刷する場合は、個々の画素ではドットを形成するか否かの 2 通りしか取り得ず、例えばドットの大きさなどを変えたとしても、画素あたりに表現可能な階調数は高々数階調に過ぎない。換言すれば、ドットを形成して画像を印刷する場合、画像データと同じ解像度で印刷すると、個々の画

素が有する画像データの情報量が大きく失われている。これに対して、画像を印刷する際に、1つの画素を複数の画素に分割してドットを形成してやれば、画像データの個々の画素が有する情報量をドットの形成状態に反映させることが可能となり、印刷画質が向上するのである。

#### 【0153】

第1の変形例の画像印刷処理は、このように、低解像度の画像データを受け取って、見かけ上の解像度を高解像度化した後に画像を印刷する場合に好適に適用することが可能である。すなわち、画像データを受け取ると、必要に応じて解像度を調整した後、個々の画素をあたかも画素群のように扱って個数データを生成する。こうすれば、受け取った画像データの解像度を高解像度化することなく、印刷解像度の画像を迅速に印刷することが可能となる。特に、受け取った画像データの画素の大きさが、画素群の大きさと一致している場合には、解像度を調整することなく、そのまま各画素の個数データを生成することができるので、より一層速やかに画像を印刷することが可能である。

#### 【0154】

B-8. 第2の変形例：

上述した第1実施例のドット形成有無決定処理においては、予め1組または複数組の順序値マトリックスを設定しておき、常に同じ順序値マトリックスを参照して、あるいは参照する順序値マトリックスをランダムに切り換えながら、各画素についてのドット形成の有無を決定するものとして説明した。しかし、図8および図9を用いて前述したように、順序値マトリックスをディザマトリックスに基づいて生成し、画素群の位置に応じて適切な順序値マトリックスを参照しながらドット形成の有無を決定してやれば、より適切にドット形成の有無を決定することができ、従って高画質な画像を印刷することが可能となる。以下では、こうした第1実施例における第2の変形例としてのドット形成有無決定処理について説明する。

#### 【0155】

図23は、第2の変形例としてのドット形成有無決定処理の流れを示すフローチャートである。以下、フローチャートに従って説明する。第2の変形例としてのドット形成有無決定処理においても、先ず初めに画素群を1つ選択し（ステップS500）、その画素群の個数データを取得する（ステップS502）。次いで、複数組記憶されている順序値マトリックスの中から、選択した画素群に対応する順序値マトリックスを読み込む（ステップS504）。かかる処理について、図24および図25を参照しながら詳しく説明する。

#### 【0156】

図24は、第2の変形例において、ドット形成有無決定処理中で参照される複数の順序値マトリックスを生成する方法について示した説明図である。ここでは、複数の順序値マトリックスをディザマトリックスに基づいて生成する。先ず、ディザマトリックスを画素群と同じ大きさを有する複数のブロックに分割して、各ブロックに通し番号を付しておく。図24(a)は、ディザマトリックスを複数のブロックに分割した様子を概念的に示した説明図である。今、ディザマトリックスが図6に示した大きさ（すなわち、主走査方向に128画素、副走査方向に64画素）を有しているものとする、1つの画素群は主走査方向に4画素、副走査方向に2画素の大きさを有するとしているから、図24(a)に示すように、ディザマトリックスは主走査方向および副走査方向にそれぞれ32ブロックずつ、全体では、1番から1024番までの通し番号が付された1024個のブロックに分割されることになる。

#### 【0157】

尚、前述した第1実施例の個数データ生成処理では、図6に示したディザマトリックスを想定して画素群に分類番号を付与するとともに（図12参照）、個数データを求めて第1の変換テーブルに設定した（図15参照）。第2の変形例のドット形成有無決定処理においても、具体的に説明するために、図6に示したディザマトリックスを想定して順序値マトリックスを生成するものとして説明するが、これらディザマトリックスは必ずしも同



一のマトリックスである必要はなく、更には、異なる大きさのディザマトリックスを用いることも可能である。

#### 【0158】

図24(a)に示すように、ディザマトリックスを通し番号1番から1024番までの複数のブロックに分割したら、各ブロックから1組ずつ順序値マトリックスを生成する。図24(b)は、一例として、通し番号1番のブロックから順序値マトリックスを生成している様子を示した説明図である。図24(b)の左側半分には、通し番号1番のブロックに含まれるディザマトリックスの閾値が示されている。図7を用いて前述したように、ディザ法では画像データの階調値とディザマトリックスの閾値とを比較して、画像データの方が大きい場合にドットを形成すると判断しているから、ディザマトリックスの閾値が小さい画素ほどドットが形成され易くなる。従って、図24(b)に示した1番のブロックの中で1番初めにドットが形成される画素は、閾値「1」が設定された画素と考えることができる。そこで、この画素には順序値として「1」を設定する。同様に、2番目にドットが形成される画素は、2番目に小さな閾値である閾値「42」が設定された画素と考えることができる。そこで、この画素には順序値「2」を設定する。このようにして、ブロック内に設定されている閾値の小さな画素から順番に、順序値「1」から順序値「8」までを決定してやれば、図24(b)の右側半分に示した通し番号1番の順序値マトリックスを得ることができる。

#### 【0159】

図24(c)は、同様にして、ブロック内で小さな閾値が設定されている画素から順番に、順序値「1」から順序値「8」までを設定することで、通し番号2番の順序値マトリックスを生成した様子を示している。図24(a)に示した通し番号「1」番から通し番号「1024」番までの全てのブロックについて、以上のような操作を行うことにより、通し番号「1」番から「1024」番までの順序値マトリックスを生成して記憶しておく。図23のステップS404では、これら「1」番から「1024」番の順序値マトリックスの中から、ドット形成の有無を決定しようとする画素群に対応するマトリックスを選択して読み込む処理を行う。

#### 【0160】

図25は、画素群に対応する順序値マトリックスを選択する方法を示した説明図である。今、ドット形成の有無を決定しようとしている画素群が、図25(a)に示すように、画像の一番左上隅を基準として主走査方向に*i*個目の画素群、副走査方向に*j*個目の画素群の位置にあるとする。また、このような画素群の位置を、座標値(*i*, *j*)によって表すものとする。また、ディザマトリックスの大きさは、通常は、画像のようには大きくはないので、図13(b)を用いて前述したように、ディザマトリックスを主走査方向に移動させながら、繰り返して使用するものとする。

#### 【0161】

図24(a)に示したように、ディザマトリックスを分割するブロックの大きさは、個数データを生成した画素群の大きさと一致しているから、ディザマトリックスを主走査方向に移動させながら繰り返し使用すると、ディザマトリックスの各ブロックは画素群の位置に一致する。換言すれば、全ての画素群には、ディザマトリックスを分割するいずれかのブロックが適用されることになる。

#### 【0162】

今、処理しようとしている画素群には、ディザマトリックス中で主走査方向に*I*個目、副走査方向に*J*個目のブロックが適用されたものとする。図24(a)に示すように、ここでは、1つのディザマトリックスには主走査方向・副走査方向にそれぞれ32個ずつのブロックが含まれるとしており、また、処理しようとする画素群の座標値は(*i*, *j*)、すなわち、画像の左上隅を基準として主走査方向に*i*個目、副走査方向に*j*個目の位置にあるとしているから、*I*、*J*はそれぞれ次式で求めることができる。

$$\begin{aligned} I &= i - \text{int}(i/32) \times 32 \\ J &= j - \text{int}(j/32) \times 32 \end{aligned}$$

ここで、 $\text{int}$  は、小数点以下を切り捨てて整数化することを表す前述した演算子である。従って、ある画素群についてドット形成の有無を決定する場合には、画素群の座標値 ( $i, j$ ) から上式によって  $I, J$  を求めた後、ディザマトリックス中で対応する位置にあるブロックの通し番号を取得して、そのブロックから生成された順序値マトリックスを用いればよい。

#### 【0163】

もちろん、実際には、図 25 (c) に示すような計算を実行せずとも、 $i, j$  の 2 進数データに対してビット毎の論理積を算出することで、それぞれ  $I, J$  の値を極めて簡便に求めることができる。以下、図 26 を参照しながら簡単に説明する。

#### 【0164】

図 26 は、画素群の座標値 ( $i, j$ ) から、適用する順序値マトリックスを選択する方法を具体的に示した説明図である。図 26 (a) は、数値  $i$  を表す 10 ビットの 2 進数データを概念的に示している。図 26 (a) では、各ビットを識別するために、最上位ビットから最下位ビットに向かって 1 番から 10 番までの通し番号を付して表示している。

#### 【0165】

順序値マトリックスの選択に際しては、先ず初めに、 $\text{int}(i/32)$  を算出する。図 14 を用いて前述したように、この演算は、 $i$  の 2 進数データを右方向に 5 ビット分だけビットシフトさせることで実行することができる (図 26 (b) 参照)。次いで、 $\text{int}(i/32) \times 32$  を算出する。この演算は、 $\text{int}(i/32)$  の 2 進数データを左方向に 5 ビット分だけビットシフトさせることで実行することができる (図 26 (c) 参照)。最後に、数値  $i$  から、 $\text{int}(i/32) \times 32$  を減算すれば、目的とする数値  $I$  を得ることができる。この操作は、結局は、数値  $i$  の 2 進数データから下位の 5 ビットのみを抜き出していることに他ならないから、図 26 (a) に示した数値  $i$  の 2 進数データと、図 26 (e) に示すマスクデータとの論理積を求めるだけで、極めて簡便に数値  $I$  を得ることが可能である。同様にして、数値  $j$  の 2 進数データと図 26 (e) のマスクデータとの論理積を求めれば、数値  $J$  を算出することができる。このように、画素群の座標値 ( $i, j$ ) が与えられれば、数値  $i, j$  から数値  $I, J$  を求めることにより、その画素群には通し番号が何番の順序値マトリックスが適用されるかを知ることができるのである。図 23 に示したドット形成有無決定処理のステップ S504 では、このようにして、画素群に対応するマトリックスを選択して読み込む処理を行う。

#### 【0166】

以上のようにして、画素群に対応する順序値マトリックスを読み込んだら、処理中の画素群の中から、ドット形成の有無を決定しようとする対象画素を 1 つ選択する (ステップ S506)。そして、読み込んでおいた順序値マトリックスを参照することで、対象画素の順序値を取得した後 (ステップ S508)、第 2 の変換テーブルを参照することによって、対象画素についてのドット形成の有無を決定する (ステップ S510)。図 19 の例示を用いて前述したように、第 2 の変換テーブルには、順序値と個数データとの組合せに対応付けて、ドット形成の有無が設定されている。従って、画素群の個数データと対象画素の順序値とが分かれば、変換テーブルを参照することで直ちにドット形成の有無を決定することができる。

#### 【0167】

こうして、1 つの対象画素についてドット形成の有無を決定したら、選択した画素群内の全画素について、ドット形成の有無を決定したか否かを判断する (ステップ S512)。未だ、ドット形成の有無を決定していない画素が残っていれば (ステップ S512: no)、ステップ S506 に戻って、新たな対象画素を選択し、続く一連の処理を行う。こうした処理を繰り返し、画素群内の全画素についてドット形成の有無を決定したと判断されたら (ステップ S512: yes)、今度は、個数データが供給された全画素群について処理を終了したか否かを判断する (ステップ S514)。そして、未処理の画素群が残っていれば (ステップ S514: no)、ステップ S500 に戻って新たな画素群を選択し、続く一連の処理を行う。こうした処理を繰り返し、全ての画素群について処理が終了

したら（ステップ S514：yes）、図 23 に示した第 2 の変形例におけるドット形成有無決定処理を終了して、図 5 の画像印刷処理に復帰する。

#### 【0168】

以上に説明した第 2 の変形例におけるドット形成有無決定処理では、ディザマトリックスに基づいて複数の順序値マトリックスを生成する。そして、ある画素群についてドット形成の有無を決定する際には、ディザ法を適用したときに、その画素群の位置に適用される部分のディザマトリックスから生成された順序値マトリックスを用いて、ドット形成の有無を決定する。こうすれば、ディザマトリックスを用いて得られたドットの分布と、ほぼ同じような分布が得られるように、ドット形成の有無を決定することができる。周知のように、ディザマトリックスには、ドットが適切な分布で形成されるように、適切な分布で閾値が設定されているから、ディザマトリックスによるドットの分布に準じた分布が得られれば、高画質な画像を印刷することが可能となる。

#### 【0169】

更に、順序値マトリックスを生成するために用いるディザマトリックスと、図 11 に示した個数データ生成処理中で使用するディザマトリックスとを同じマトリックスとしておけば、図 8 および図 9 を用いて前述したように、個数データから復元したドット分布は、ほとんどの場合は、ディザ法を用いて画素毎にドット形成の有無を判断した場合と全く同じドット分布となる。もちろん、前述したように、画素群内で画像データの階調値が大きく変化している場合には、ドット分布は異なったものとなるが、画像データには隣接する画素間では近似した（若しくは同一の）階調値を有する傾向があるため、多くの場合、ドット分布は同一となる。従って、適切なドット分布となるようにドット形成の有無を決定することができ、それだけ高画質な画像を印刷することが可能となる。

#### 【0170】

B-9. メモリ使用量：

以上に説明した第 1 実施例の画像印刷処理によれば、個数データ生成処理では、第 1 の変換テーブルを参照することによって個数データを求めている。このため、ディザマトリックスを記憶しておく必要は無い。もっとも、ディザマトリックスの代わりに第 1 の変換テーブルを記憶しておく必要がある。また、ドット形成有無決定処理では、順序値マトリックスおよび第 2 の変換テーブルを記憶しておく必要がある。更に、上述した第 2 の変形例においては、順序値マトリックスを複数記憶しておく必要がある。このため、これら変換テーブルや順序値マトリックスのデータサイズがあまりに大きくなったのでは、メモリ使用量の点から制約を受けることになる。しかし、以下に説明するように、第 1 変換テーブル、第 2 の変換テーブル、および順序値マトリックスのデータサイズは決して大きなものではなく、メモリ使用量の点から制約を受けることはない。

#### 【0171】

図 27 は、第 1 の変換テーブルを記憶するためのメモリ使用量を、各種条件下で試算した結果を示す説明図である。図 15 を用いて前述したように、第 1 の変換テーブルには、分類番号および画素群階調値の組合せ毎に個数データが設定されているから、第 1 の変換テーブルのデータサイズは、分類番号の個数と、画素群階調値が取り得る範囲と、個数データ 1 つ当たりのデータ長とによって決定される。また、分類番号の個数は、ディザマトリックスを分割して生成したブロック数と等しいから、結局、ディザマトリックスのサイズと画素群の大きさによって決定される。図 27 では、各種サイズのディザマトリックスと、各種大きさの画素群とを想定し、これらを組合せた時の変換テーブルのデータ量が試算示されている。具体的には、ディザマトリックスの大きさとしては、 $64 \times 64$ （すなわち主走査方向に 64 画素、副走査方向に 64 画素）、 $128 \times 64$ （主走査方向に 128 画素、副走査方向に 64 画素）、 $128 \times 128$ （主走査方向に 128 画素、副走査方向に 128 画素）の 3 種類のサイズを想定している。画素群のサイズとしては、 $2 \times 2$ （主走査方向に 2 画素、副走査方向に 2 画素）、 $4 \times 2$ （主走査方向に 4 画素、副走査方向に 2 画素）、 $4 \times 4$ （主走査方向に 4 画素、副走査方向に 4 画素）の 3 種類のサイズを想定している。また、ここでは、画素群階調値は 0～255 の 256 通りの値を取り得る

ものとし、個数データのデータ長は1バイトであるものとしている。

#### 【0172】

上述した実施例に対応する条件、すなわち、ディザマトリックスサイズが $128 \times 64$ であり、画素群の大きさが $4 \times 2$ の条件での試算結果は、図27では、破線で囲って示されている。以下では、この条件を代表例として用いながら、第1の変換テーブルを記憶するために使用するメモリ量の試算結果について説明する。ブロック数は、ディザマトリックスを画素群と同じサイズで分割して得られるブロックの個数であるから、ディザマトリックスの画素数( $128 \times 64$ )を画素群当たりの画素数( $4 \times 2$ )で除算して、1024となる。この値が、第1の変換テーブルの分類番号の個数となる。ここでは、画素群階調値は0~255までの256通りの値を取り得るとしているから、分類番号と画素群階調値との組合せは、 $1024 \times 256$ 通り存在している。この組合せ毎に、1バイトのデータ長の個数データが記憶されるから、結局、第1の変換テーブルのデータ量は、256Kバイトと算出される。

#### 【0173】

上述した算出方法から明らかなように、第1の変換テーブルを記憶するために使用するメモリ量は、ディザマトリックスのサイズが大きくなって分類番号の個数が増えるほど大きくなる傾向にある。同様に、画素群の大きさが小さくなるほど分類番号の個数が増えるので、第1の変換テーブルのデータ量は大きくなる傾向にある。ここで、実際に用いられるディザマトリックスのサイズは、 $64 \times 64 \sim 128 \times 128$ 程度であるから、図27に示した試算結果から、第1の変換テーブルのデータ量は、極めて特殊な場合を除いて1Mバイトを越えることはなく、256Kバイト~512Kバイト程度に収まっている。もちろん、図27の試算で用いたディザマトリックスよりも大きなマトリックスを使用する場合もあるが、その場合でも、第1の変換テーブルのデータ量はメモリ容量を圧迫するほど大きなものにはならないと考えられる。

#### 【0174】

また、図27に示した試算では、個数データ1つ当たりのデータ長は1バイトであるとしたが、実際には1バイトものデータ長は不要であり、これを考慮すれば、第1の変換テーブルのデータ量は更に小さなものとなる。この点について、再び図27を参照しながら説明する。図中で、破線で囲った条件(上述した実施例の条件)について説明すると、画素群内の画素数は8つであるから、画素群当たりに形成されるドット個数は0個~8個までの合計9つの状態を取り得る。9つの状態であれば4ビットあれば表現することができるから、個数データ1つあたりのデータ長は4ビットあれば良く、従って、第1の変換テーブルのデータ量も128Kバイトと半減する。図27で、右端から2列目の「状態数/画素群」と表示された欄には、画素群が取り得る状態数(すなわち1つの画素群に形成され得るドット個数の種類)が示されており、右端の「使用ビット数」と表示された欄には、状態数を表現するために必要なビット数(すなわち、個数データ1つ当たりのデータ長)が示されている。

#### 【0175】

前述したように、個数データが1バイトデータであるとした場合は、画素群の大きさが小さくなるほど変換テーブルのデータ量は大きくなる。しかし、図27の右端の欄に示されているように、画素群の大きさが小さくなるほど使用ビット数は小さくなるので、この点を考慮したときのデータ量の低減率は大きくなる。実際、画素群の画素数が8つの場合は、使用ビット数を考慮することでデータ量が半減するが、画素群の画素数が4つの場合はデータ量は $3/8$ と、4割以下に減少する。こうした点を考慮すれば、第1の変換テーブルを記憶するために使用するメモリ量は図27の試算結果よりも更に減少し、実際には高々256Kバイト程度あれば十分であると考えられる。

#### 【0176】

前述した画像印刷処理中の色変換処理(図5のステップS102)で参照する色変換テーブル(LUT)のデータ量が、通常は1.5M以上になることと比べれば、256Kバイトというデータ量は決して大きなデータ量ではない。むしろ、現在の一般的なコンピュ

ータのキャッシュメモリにも十分に収まる程度の小さなデータ量とすることができる。従って、コンピュータ100で個数データ生成処理を実行する場合、第1の変換テーブルのデータ量がメモリ容量を圧迫することはあり得ないと考えられる。また、コンピュータではなく、デジタルカメラ120などの画像機器やカラープリンタ200の内部で個数データ生成処理を実施する場合も、256Kバイト程度であれば、特別にメモリを追加せずとも既存のメモリで吸収することが可能である。特に、現在の一般的なプリンタでは、プリンタ内部の処理を実行するために10Mバイト程度のメモリを搭載していることから、256Kバイト程度の変換テーブルを記憶することでメモリ容量が圧迫されることはない。

#### 【0177】

次に、順序値マトリックスを記憶するために必要なメモリ量について説明する。順序値マトリックスを記憶するためのメモリ量は、マトリックス1つ当たりには要するメモリ量と、マトリックスの個数とによって決定される。順序値マトリックスには、画素群内の各画素にドットが形成される順番が設定されているから、マトリックス1つ当たりのメモリ量は、画素群に含まれる画素数によって決まる。また、順序値マトリックスの個数は、図24を用いて前述したように、ディザマトリックスを画素群と同じ大きさのブロックで分割した時に得られるブロック数に等しいから、ディザマトリックスの大きさと、画素群の大きさとによって決定される。結局、順序値マトリックスを記憶するために必要なメモリ量は、ディザマトリックスの大きさと、画素群の大きさとによって決定されることになる。

#### 【0178】

図28は、第1の変換テーブルについて試算したものと同様の各種条件を想定して、順序値マトリックスを記憶するために必要なメモリ量を試算した結果を示す説明図である。また、上述した実施例に対応する条件、すなわち、ディザマトリックスサイズが128×64であり、画素群の大きさが4×2の条件での試算結果は、図28では、破線で囲って示されている。以下では、この条件を代表例として用いながら、順序値マトリックスの試算例について説明する。

#### 【0179】

順序値マトリックスの個数は、ディザマトリックスを画素群と同じサイズで分割して得られるブロックの個数であるから、ディザマトリックスの画素数(128×64)を画素群当たりの画素数(4×2)で除算して、1024となる。また、順序値マトリックスに設定される順序値は、1～8までの8通りの値を取るから、1つの順序値は3ビットあれば表現することができる。順序値マトリックスには8つの順序値が設定されているから、1つの順序値マトリックスを記憶するために必要なメモリ量は、3×8=24ビット(3バイト)となる。順序値マトリックスの個数は1024個あるから、全ての順序値マトリックスを記憶するために必要なメモリ量は、3Kバイトと求めることができる。

#### 【0180】

また、画素群に含まれる画素数が4つの場合は、順序値は1～4までの4通りの値を取るから、1つの順序値は2ビットあれば表現することができる。順序値マトリックスには4つの順序値が設定されるから、1つの順序値マトリックスを記憶するために要するメモリ量は、2×4=8ビット(1バイト)となる。同様に、画素群に含まれる画素数が16個の場合は、1つの順序値を表現するために要するデータ長は4ビット、これが16個あるから、順序値マトリックスを記憶するために要するメモリ量は、4×16=64ビット(8バイト)となる。

#### 【0181】

図28には、このようにして、各種条件で全ての順序値マトリックスを記憶するために必要なメモリ量を試算した結果が示されている。図示されている試算結果を見れば明らかなように、順序値マトリックスを記憶しておくために必要なメモリ量は、高々10Kバイトあれば十分と考えられる。このため、順序値マトリックスを記憶するために必要なメモリ量が、実際の製品に搭載する上で障害になるほど大きくなることはない。

#### 【0182】

最後に、第2の変換テーブルを記憶するために必要なメモリ容量について説明する。図

19に示したように、第2の変換テーブルには、順序値および個数データの組合せ毎にドット形成の有無が設定されている。画素群に含まれる画素数を8個とすれば、順序値は8通りの値を取る。また、個数データは画素群内に形成され得るドットの個数を表しているから、画素数+1通りの値を取り得る。更に、ここでは、1つの画素にはドットが形成されるか否かのいずれかの状態しか取り得ないとしているから、1画素分のドット形成の有無は1ビットあれば表現することができる。結局、第2の変換テーブルを記憶するためのメモリ容量は、画素群に含まれる画素数を $n$ とすると、

$$n \times (n + 1) \text{ ビット}$$

あればよい。画素群に含まれる画素数は、高々16個程度だから、第2の変換テーブルを記憶するために要するメモリ量は僅かなものである。

#### 【0183】

以上に説明したように、第1の変換テーブルや、第2の変換テーブル、複数の順序値マトリックスのデータサイズは決して大きなものではなく、これらテーブルやマトリックスを記憶しておくためのメモリ量が、実際の製品に搭載する上で障害になるほど大きくなることはない。

#### 【0184】

##### C. 第2実施例:

以上に説明した第1実施例では、カラープリンタ200で形成可能なドットは1種類であるものとして説明した。しかし、今日では、印刷画質を向上させることを目的として、大きさの異なるドットや、インク濃度の異なるドットなど、多種のドットを形成可能なプリンタ（いわゆる多値ドットプリンタ）が広く使用されている。本願の発明は、こうした多値ドットプリンタに適用した場合にも、大きな効果を得ることができる。以下では、第2実施例として、本願発明を多値ドットプリンタに適用した場合について説明する。

#### 【0185】

##### C-1. 第2実施例の画像印刷処理の概要 :

第2実施例の画像印刷処理は、フローチャートについては、図5に示した第1実施例の画像印刷処理と同様である。以下では、図5のフローチャートを流用しながら、第2実施例の画像印刷処理の概要について簡単に説明する。

#### 【0186】

第2実施例の画像印刷処理を開始すると、先ず初めに、コンピュータ100で画像データを読み込んだ後、色変換処理を行う（図5のステップS100およびステップS102相当）。次いで、解像度変換処理を行って、画像データの解像度を印刷解像度に変換した後（ステップS104相当）、個数データ生成処理を開始する（ステップS106相当）。

#### 【0187】

前述したように、第1実施例では、カラープリンタ200が形成可能なドットは1種類であるものとしており、個数データ生成処理では、画素群内に形成されるドット個数を表す個数データを画素群毎に生成して、カラープリンタ200に出力した。これに対して、第2実施例では、カラープリンタ200は、大きさの異なる3種類のドット、すなわち大ドット、中ドット、小ドットを形成可能であるものとする。このことと対応して、第2実施例の個数データ生成処理では、画素群内に、大ドット、中ドット、小ドットがそれぞれ何個ずつ形成されるかを表す個数データを生成することになる。また、詳細には後述するが、個数データを少ないデータ量で効率よく出力するために、大ドット、中ドット、小ドットの個数をそのまま出力するのではなく、コード化された状態で出力する。第2実施例の個数データ生成処理の詳細については後述する。

#### 【0188】

カラープリンタ200の制御回路260に内蔵されたCPUは、コンピュータ100から供給された個数データを受け取ると、ドット形成有無決定処理を開始する（図5のステップS108相当）。詳細には後述するが、第2実施例のドット形成有無決定処理では、コード化された状態で供給された個数データを、大ドット、中ドット、小ドットの個数を



示すデータに復号した後、これら各種ドットについてドット形成の有無を判断する。

#### 【0189】

こうして、大中小の各種ドットについてドット形成の有無を決定したら、決定したドット形成の有無に従って、印刷媒体上にドットを形成する（図5のステップS110相当）。こうして大ドット、中ドット、小ドットを形成することにより、画像データに対応した画像が印刷される。

#### 【0190】

C-2. 第2実施例の個数データ生成処理：

次に、上述した第2実施例の画像印刷処理において、画素群内に形成される大ドット、中ドット、小ドットの個数がコード化された個数データを生成する処理について説明する。後述するように、コード化された個数データも、画素群の分類番号と画素群階調値とに基づいて変換テーブルを参照することにより、極めて容易に生成することができる。こうしたことが可能である理由を説明するために、先ず初めに、画素群内に形成される大中小ドットの個数を、ディザ法を用いて決定する処理について簡単に説明する。次いで、大中小ドットの個数をコード化する処理について説明し、その後、第2実施例の個数データ生成処理で行われる詳細な処理内容について説明する。

#### 【0191】

C-2-1. ディザ法を用いた大中小ドットの形成個数の決定処理：

図29は、画素群内に形成される大ドット、中ドット、小ドットの個数を、ディザ法を適用して決定する処理の流れを示すフローチャートである。尚、かかる処理の詳細については、特許3292104号に開示されている。大中小ドットの個数を決定する場合も、処理を開始すると先ず初めに、互いに隣接する所定数の画素をまとめて画素群を形成する（ステップS600）。ここでは、前述した実施例と同様に、主走査方向に4画素、副走査方向に2画素の合計8つの画素を画素群としてまとめるものとする。

#### 【0192】

次いで、画素群の中からドット形成の有無を判断するべく、処理対象とする画素を1つ選択して（ステップS602）、選択した処理画素について、大ドット、中ドット、小ドットの形成有無を判断する（ステップS604）。大中小ドットの形成有無は次のようにして判断する。

#### 【0193】

図30は、選択した1つの画素についてハーフトーン処理を行うことにより、大ドット、中ドット、小ドットの形成有無を判断する処理の流れを示すフローチャートである。大中小ドットのハーフトーン処理を開始すると、先ず初めに処理対象とする画素についての画像データを、大ドット、中ドット、小ドットの各ドットについての密度データに変換する（ステップS650）。ここで、密度データとは、ドットをどの程度の密度で形成するかを表すデータである。密度データは、大きな階調値となるほどドットが高い密度で形成されることを表している。例えば、密度データの階調値「255」は、ドットの形成密度が100%、すなわち全ての画素にドットが形成されることを表しており、密度データの階調値「0」は、ドットの形成密度が0%、すなわちいずれの画素にもドットが形成されないことを表している。こうした密度データへの変換は、ドット密度変換テーブルと呼ばれる数表を参照することによって行うことができる。

#### 【0194】

図31は、画像データの階調値を大中小各ドットについての密度データに変換する際に参照されるドット密度変換テーブルを概念的に示した説明図である。図示されているように、ドット密度変換テーブルには、色変換によって得られた画像データの階調値に対して、小ドット・中ドット・大ドットの各ドットについての密度データが設定されている。画像データが階調値「0」近傍の領域では、中ドット・大ドットの密度データはいずれも階調値「0」に設定されている。小ドットの密度データは、画像データの階調値が大きくなるに連れて増加して行くが、画像データがある階調値に達すると今度は逆に減少し始め、代わりに中ドットの密度データが増加し始める。画像データの階調値が更に増加して、あ



る階調値に達すると、小ドットの密度データが階調値「0」となり、中ドットの密度データが減少し始めて、代わりに大ドットの密度データが少しずつ増加していく。図30のステップS650では、このドット密度変換テーブルを参照しながら、画像データの階調値を、大ドットの密度データ、中ドットの密度データ、小ドットの密度データに変換する処理を行う。

#### 【0195】

このようにして、処理対象とする画素について、大中小各ドットの密度データが得られたら、先ず初めに大ドットについての形成有無を判断する（図30のステップS652）。かかる判断は、大ドットの密度データと、処理対象としている画素の対応する位置に設定されているディザマトリックスの閾値とを比較することによって行う。そして、密度データの方が大きい場合は処理対象の画素には大ドットを形成するものと判断される。その結果、大ドットを形成することになった場合は、ステップS654において「yes」と判断され、ハーフトーン処理を抜けて図29に示したドット個数決定処理に復帰する。

#### 【0196】

逆に、大ドットの密度データよりも閾値の方が大きければ、ステップS652において、処理対象の画素には大ドットは形成されないと判断される。判断の結果、大ドットは形成されない場合は（ステップS654：no）、今度は中ドットについて形成有無を判断する処理を開始する。中ドットの形成有無の判断には、大ドットの密度データと中ドットの密度データとを加算して、中ドット用の中間データを算出する（ステップS656）。そして、得られた中ドット用の中間データと、ディザマトリックスの閾値とを比較することにより、中ドットの形成有無を判断する（ステップS658）。そして、中ドット用の中間データの方が大きい場合は処理対象の画素には中ドットを形成するものと判断される。その結果が、中ドットを形成することになった場合は、ステップS660において「yes」と判断されるので、ハーフトーン処理を抜けて図29のドット個数決定処理に復帰する。

#### 【0197】

逆に、中ドット用の中間データよりも閾値の方が大きかった場合は、ステップS658において、処理対象の画素には中ドットを形成しないと判断される。判断の結果、中ドットも形成されない場合は（ステップS660：no）、今度は小ドットについて形成有無を判断する処理を開始する。小ドットの形成有無の判断には、中ドット用の中間データと小ドットの密度データとを加算して、小ドット用の中間データを算出する（ステップS662）。そして、得られた小ドット用の中間データと、ディザマトリックスの閾値とを比較することにより、小ドットの形成有無を判断する（ステップS664）。その結果、小ドット用の中間データの方が大きい場合は処理対象の画素には小ドットを形成するものと判断し、逆に、小ドット用の中間データよりも閾値の方が大きい場合には、いずれのドットも形成されないものと判断する。以上のような処理を行えば、処理対象としている画素について、大ドット、中ドット、小ドットのいずれのドットを形成するか、若しくは、いずれのドットも形成しないかを判断することができるので、図30に示したハーフトーン処理を抜けて図29のドット個数決定処理に復帰する。

#### 【0198】

上述した処理を行いながら大中小の各ドットの形成有無を判断する様子について、図32を参照しながら補足して説明する。図32は、画素群内の各画素について、ディザ法を適用しながら大中小各ドットの形成有無を判断している様子を概念的に示した説明図である。ここでは、説明が煩雑となることを避けるために、画素群内の全画素が同じ階調値を有しており、従って、大中小各ドットの密度データも同じ階調値を有しているものとする。図32（a）は、画素群内の各画素について得られた大中小ドットの密度データを示しており、いずれの画素も、大ドットの密度データが「2」、中ドットの密度データが「90」、小ドットの密度データが「32」であったものとする。

#### 【0199】

図32（b）は、ディザマトリックス中で、画素群の対応する位置に記憶されている閾

値を表している。大ドットの形成有無を判断する際には、大ドットの密度データと、これら閾値とを比較する。ここでは、いずれの画素についても、大ドットの密度データは「2」であるとしているから、大ドットを形成すると判断される画素は、閾値「1」が設定された画素だけである。図32(b)には、大ドットが形成されると判断された画素には、細かい斜線を付して表示している。その他の画素については、中ドットか小ドットのどちらかが形成されるか、若しくはいずれのドットも形成されないかのいずれかであると考えられる。そこで、中ドットの形成有無を判断する。

#### 【0200】

中ドットの形成有無の判断に際しては、大ドットの密度データ「2」と中ドットの密度データ「90」とを加算して中ドット用の中間データを算出し、得られた中間データ「92」とディザマトリックスの閾値とを比較する。その結果、閾値「42」が設定された画素と、閾値「58」が設定された画素の2つの画素にのみ、中ドットが形成されるものと判断される。図32(c)には、中ドットが形成されると判断された画素には、少し細かい斜線を付して表示している。そして、大ドットも中ドットも形成されない画素については、小ドットが形成されるか、ドットが形成されないかのいずれかであると考えられる。そこで、中ドット用の中間データ「92」に小ドットの密度データ「32」を加算して、小ドット用の中間データを算出し、得られた中間データ「124」とディザマトリックスの閾値とを比較する。その結果、閾値「109」が設定された画素にのみ、小ドットが形成されるものと判断される。図32(d)には、小ドットが形成されると判断された画素には、粗い斜線を付して表示している。

#### 【0201】

図29に示したドット個数決定処理のステップS602～S606では、以上のようにして画素群内の各画素について中間データを算出しながら、大中小の各ドットについての形成有無を判断していく。こうして、画素群内の全画素について判断を終了したら（ステップS606：yes）、画素群内に形成される大ドット、中ドット、小ドットの個数を取得する（ステップS608）。図32に例示した画素群については、大ドット1個、中ドット2個、小ドット1個となる。

#### 【0202】

こうして、大中小の各ドットのドット個数を取得したら、画像の全画素について以上の処理を行ったか否かを判断する（ステップS610）。そして、未処理の画素が残っている場合は、ステップS600に戻って続く一連の処理を繰り返し、画像の全画素について処理を終了したと判断されたら、図29に示したディザ法によるドット個数決定処理を終了する。その結果、画像データは複数の画素群に分割され、各画素群に形成される大ドット、中ドット、小ドットの個数が得られることになる。図33は、画素群毎に大ドット、中ドット、小ドットの形成個数が得られた様子を概念的に示した説明図である。

#### 【0203】

C-2-2. 大中小ドットの形成個数のコード化処理 :

以上のようにして画素群毎に各ドットの個数が求められたら、求めたドットの個数をコード化した状態でプリンタに出力する。すなわち、例えば図33に示すように、ドットの種類が大中小の3種類である場合、ドットの種類毎に形成する個数を出力したのでは、1つの画素群について3回ずつドット個数を出力しなければならない。これでは、コンピュータ100からカラープリンタ200にデータを迅速に出力することで画像を迅速に印刷するという効果が減殺されてしまう。そこで、各ドットの個数を個別に出力するのではなく、各ドットの個数の組合せ、例えば（大ドットがK個、中ドットがL個、小ドットがN個）という組合せを、組合せ毎に設定された個別のコードに変換した後、得られたコードを出力するのである。

#### 【0204】

大中小各ドットの組合せをコード化する処理は、ドット個数の組合せとコードデータとを対応付けた状態で予め対応テーブルに記憶しておき、この対応テーブルを参照することで行う。図34は、画素群に形成される大中小各ドットの個数の組合せと、コードデータ

とが対応付けて設定された対応テーブルを示す説明図である。図34に例示の対応テーブルには、例えば大ドット、中ドット、小ドットの個数がいずれも0個である組合せには、コードデータ「0」が対応付けられている。また、大ドットが0個、中ドットが0個、小ドットが1個の組合せには、コードデータ「1」が対応付けられている。このように対応テーブルには、各ドットの個数の組合せ毎に、予め固有のコードデータが対応付けて設定されている。

#### 【0205】

ここで、大中小ドットの個数の組合せ数は、次のようなものとなる。画素群内の各画素には、大ドット、中ドット、小ドットのいずれのドットも形成され得るが、1つの画素に複数のドットが形成されることはないから、ドット個数の合計が画素群内の画素数（上述した実施例では8個）を越えることはない。従って、これら大中小ドットの個数の組合せは、「大ドットを形成する」、「中ドットを形成する」、「小ドットを形成する」、「ドットを形成しない」の4つの状態の中から重複を許して8回選択するときの組合せの数に等しくなるから、

$${}_4H_8 \quad (= {}_{4+8-1}C_8)$$

によって求められ、165通りの組合せが存在していることになる。ここで、 ${}_nH_r$  は、 $n$ 種類の物の中から重複を許して $r$ 回選択したときに得られる組合せの数（重複組合せ数）を求める演算子である。また、 ${}_nC_r$  は、 $n$ 種類の物の中から重複を許さずに $r$ 回選択したときに得られる組合せの数を求める演算子である。

#### 【0206】

このように、大中小各ドットの個数の組合せが165通り存在していることから、コードデータも「0」～「164」の165通りあればよい。165通りであれば、8ビットのデータ長があれば表現することができる。結局、大ドットの個数、中ドットの個数、小ドットの個数と、3回出力する代わりに、8ビットのコード化された個数データを1回出力するだけで、画素群に形成する各種ドットの個数を出力することが可能となる。そこで、図33に示すような画素群毎に得られたドット個数の組合せを、図34に示すような対応テーブルを参照しながらコード化された個数データに変換してからプリンタに供給することで、個数データの供給を迅速化して画像を迅速に印刷するのである。

#### 【0207】

C-2-3. 変換テーブルを利用した第2実施例の個数データ生成処理 :

以上では、画素群内に形成する大中小ドットの個数をディザ法を用いて決定した後、得られたドット個数の組合せをコード化してからプリンタに供給するという、言わば2段階の処理を経るものとして説明した。しかし、第2実施例の個数データ生成処理では、変換テーブル（第1の変換テーブル）を参照することにより、画素群の画像データをコード化された個数データに直接変換して、カラープリンタ200に出力する。このため、個数データを極めて迅速に生成することが可能となるとともに、個数データを生成する処理もたいへん簡素なものとなる。その結果、コンピュータのような高い処理能力を有する機器を用いずとも、十分に実用的な速度で個数データを生成することが可能となる。以下では、こうした第2実施例の個数データ生成処理について説明する。

#### 【0208】

図35は、第2実施例の個数データ生成処理の流れを示すフローチャートである。尚、以下では、第2実施例の個数データ生成処理もコンピュータ100で実施されるものとして説明するが、前述した第1実施例と同様に、第2実施例の個数データ生成処理も極めて簡素な処理とすることができるので、カラープリンタ200あるいはデジタルカメラ120内で実施することも可能である。以下、フローチャートに従って説明する。

#### 【0209】

第2実施例の個数データ生成処理を開始すると、まず初めに、互いに隣接する所定個数の画素をまとめて画素群を生成する（ステップS700）。ここでは、上述した各種実施例と同様に、主走査方向に4画素分、副走査方向に2画素分の合計8つの画素を画素群にまとめるものとする。次いで、画素群の分類番号と、画素群階調値とを決定する（ステッ

プS702)。画素群の分類番号および画素群階調値の決定方法は、前述した第1実施例と同様であるため、ここでは説明は省略するが、分類番号および画素群階調値はいずれも極めて簡便に決定することができる。

#### 【0210】

次いで、画素群の分類番号と画素群階調値とから第1の変換テーブルを参照することにより、コード化された個数データを決定する（ステップS704）。図36は、第2実施例の個数データ生成処理において参照される第1の変換テーブルを概念的に示した説明図である。図示されているように、第2実施例における第1の変換テーブルには、画素群の分類番号と画素群階調値との組合せに対応付けて、コード化された個数データが予め記憶されている。従って、第2実施例の個数データ生成処理においては、画素群の分類番号および画素群階調値を決定して第1の変換テーブルを参照するだけで、直ちにコード化された状態の個数データを決定することが可能である。

#### 【0211】

以上のようにして、1つの画素群についてコード化された個数データが得られたら、画像データの全画素について処理を終了したか否かを判断する（ステップS706）。そして、未処理の画素が残っている場合は（ステップS706：no）、ステップS700に戻って新たな画素群を生成し、続く一連の処理を繰り返す。こうした操作を繰り返し、全画素についての処理が終了したと判断されたら（ステップS706：yes）、各画素群について得られたコード化済みの個数データをカラープリンタ200に出力して（ステップS708）、図35に示す第2実施例の個数データ生成処理を終了する。

#### 【0212】

##### C-3. 第2実施例のドット形成有無決定処理：

次に、第2実施例のカラープリンタ200において、コード化された個数データを受け取って大中小各ドットの形成有無を決定する処理について説明する。前述した第1実施例のドット形成有無決定処理では、第2の変換テーブルを参照することで、個数データと順序値とから直ちにドット形成の有無を決定したが、第2実施例のドット形成有無決定処理においても同様に、第2の変換テーブルを参照することにより、コード化された個数データと順序値とから直ちに大中小ドットの形成有無を決定することが可能である。このような処理が可能な理由を明らかにするとともに、こうした処理を行うことで処理が高速化されることを示すために、以下では、先ず初めに、変換テーブル（第2の変換テーブル）を用いずに個数データからドット形成の有無を決定する処理について説明する。そして、次に、第2の変換テーブルを参照することで、個数データから大中小各種ドットの形成有無を迅速に決定可能な、第2実施例のドット形成有無決定処理について説明する。

#### 【0213】

##### C-3-1. 変換テーブルを参照せずにドット形成有無を決定する方法：

図37は、第2の変換テーブルを参照することなく、大中小各種ドットの形成有無を決定する処理の流れを示すフローチャートである。以下では、フローチャートに従って簡単に説明する。処理を開始すると、先ず初めに、画素位置を決定しようとする画素群を1つ選択し（ステップS750）、その画素群の個数データを取得する（ステップS752）。こうして取得された個数データは、前述したようにデータ量を減少させるためにコード化されたデータとなっている。そこで、個数データを復号して、大ドット、中ドット、小ドットについての個数を示すデータに変換する処理を行う（ステップS754）。個数データを復号する処理は、復号テーブルを参照することによって行う。

#### 【0214】

図38は、コード化された個数データを復号するために参照される復号テーブルを概念的に示した説明図である。図示されるように、復号テーブルには、コード化された個数データに対応する大ドット、中ドット、小ドットのドット個数の組合せが設定されている。例えば、コード化された個数データが「1」である場合は、大ドットおよび中ドットの個数は0個で、小ドットの個数が1個であるドット個数の組合せに復号される。図37のステップS754では、このような復号テーブルを参照することにより、コード化された個

数データを大中小各ドットの個数を表すデータに変換する。

#### 【0215】

次いで、個数データを復号した画素群に対応する順序値マトリックスを読み込んだ後（ステップS756）、読み込んだ順序値マトリックスを参照しながら、画素群内の各画素について大ドット、中ドット、小ドットの形成有無を判断していく。ここで順序値マトリックスとは、図24を用いて前述したように、画素群内の各画素についてドットが形成される順番を設定したマトリックスである。

#### 【0216】

図39は、順序値マトリックスを参照しながら、大中小の各種ドットについての形成有無を画素毎に決定している様子を概念的に示した説明図である。例えば、個数データを復号することにより、大ドットの個数が1個、中ドットの個数が2個、小ドットの個数が1個の組合せが得られたものとする。各画素についてのドット形成の有無を決定するに際しては、先ず初めに大ドットを形成する画素を決定する（図37のステップS758）。ここでは、大ドットの個数は1個であるとしているから、最もドットの形成されやすい画素、すなわち、順序値マトリックスで順序値が「1」に設定されている画素に大ドットが形成されるものと判断する。ここで、大ドットの個数がN個である場合は、順序値マトリックスで順序値が「1」から「N」までの値が設定されている画素に大ドットを形成するものと判断する。図39では、大ドットを形成する画素には、細かい斜線を付して表示している。

#### 【0217】

大ドットを形成する画素を決定したら、大ドットが形成されなかった画素の中から、今度は、中ドットを形成する画素を決定する（図37のステップS760）。図39に示すように、中ドットの個数は2個であり、順序値「1」が設定されている画素位置には大ドットが形成されているから、中ドットは順序値「2」が設定された画素と順序値「3」が設定された画素とに形成される。図39では、中ドットが形成される画素位置には、少し粗い斜線を付して表示している。

#### 【0218】

こうして中ドットを形成する画素を決定したら、大ドットも中ドットも形成されない画素の中から、小ドットを形成する画素を決定する（図37のステップS762）。図39に示すように、小ドットの個数は1個であり、順序値「1」の画素には大ドットが、順序値「2」および「3」の画素には中ドットが形成されているから、小ドットは順序値「4」が設定された画素に形成される。図39では、小ドットが形成される画素には、粗い斜線を付して表示している。そして、最後に、大ドット、中ドット、小ドットのいずれも形成されなかった画素については、ドットを形成しないものと判断する（図37のステップS764）。

#### 【0219】

こうして、1つの画素群について、コード化された個数データを復号し、大中小の各ドットを形成する画素を決定したら、全画素群について処理を終了したか否かを判断する（図37のステップS766）。そして、未処理の画素群が残っている場合は（ステップS766：no）、ステップS600に戻って、新たな画素群について続く一連の処理を繰り返す。こうして全ての画素群について処理を終了したと判断されたら（ステップS766：yes）、図37に示したドット形成有無決定処理を終了する。

#### 【0220】

C-3-2. 変換テーブルを参照するドット形成有無決定処理：

以上に説明したドット形成有無の決定処理では、コード化された個数データを受け取ると、画素群内に形成する大中小ドットの個数を示すデータに復号した後、各画素についていずれのドットを形成するかを決定するという、言わば2段階の操作を経ている。しかし、第2の変換テーブルを参照すれば、各画素に形成するドットを、個数データを復号することなく直ちに決定することができる。以下では、第2の変換テーブルを参照してドット形成の有無を決定する第2実施例のドット形成有無決定処理について説明する。

**【0221】**

第2実施例のドット形成有無決定処理は、前述した第1実施例において図23を用いて説明した変形例のドット形成有無決定処理に対して、参照する第2の変換テーブルが異なっているだけであり、処理の流れは同様である。そこで、以下では、図23のフローチャートを流用しながら、第2実施例のドット形成有無決定処理について説明する。

**【0222】**

第2実施例のドット形成有無決定処理を開始すると、先ず初めに画素群を1つ選択し（ステップS500相当）、その画素群の個数データを取得する（ステップS502相当）。次いで、複数組記憶されている順序値マトリックスの中から、選択した画素群に対応する順序値マトリックスを読み込んでやる（ステップS504相当）。すなわち、図25および図26を用いて前述したように、画素群の座標値（ $i, j$ ）から、 $i$ および $j$ の下位5ビットを抜き出して、それぞれ $I, J$ を求める。そして、ディザマトリックス中で $I$ 行 $J$ 列のブロックから生成された順序値マトリックスを選択して、読み込んでやればよい。

**【0223】**

以上のようにして、画素群に対応する順序値マトリックスを読み込んだら、処理中の画素群の中から、ドット形成の有無を決定しようとする対象画素を1つ選択する（ステップS506相当）。そして、読み込んでおいた順序値マトリックスを参照することで、対象画素の順序値を取得した後（ステップS508相当）、第2の変換テーブルを参照することによって、対象画素についてのドット形成の有無を決定する（ステップS510相当）。

**【0224】**

ここで、前述した第1実施例のドット形成有無決定処理において参照した第2の変換テーブルには、個数データと順序値との組合せ毎に、ドット形成の有無を表すデータが設定されていた（図19参照のこと）。これに対して、第2実施例のドット形成有無決定処理で参照する第2の変換テーブルには、コード化された状態の個数データと順序値との組合せ毎に、大中小いずれのドットを形成するか、あるいはドットを形成しないかを示すデータが設定されている。

**【0225】**

図40は、第2実施例のドット形成有無決定処理において参照される第2の変換テーブルを概念的に示した説明図である。図34を用いて前述したように、コード化された個数データは0～164までの165通りの値を取り得る。また、1つの画素群は8つの画素から構成されているとしているから、順序値は1～8までの8通りの値を取り得る。第2実施例における第2の変換テーブルには、これら $165 \times 8 = 1320$ 通りの組合せ毎に、ドットを形成しないことを示す値「0」、小ドットを形成することを示す値「1」、中ドットを形成することを示す値「2」、大ドットを形成することを示す値「3」のいずれかが設定されている。従って、画素群の個数データと対象画素の順序値とが分かれば、直ちにドット形成の有無を決定することができる。第2実施例のドット形成有無決定処理では、このような第2の変換テーブルを参照することで、対象画素に大中小いずれのドットを形成するのか、あるいはドットを形成しないのかを、直ちに決定するのである（図23のステップS510相当）。

**【0226】**

ここで、図40に示すような第2の変換テーブルを参照しながらドット形成の有無を決定した場合でも、大中小ドットについてのドット形成の有無を適切に決定可能な理由について説明する。図37ないし図39を用いて前述したように、第2の変換テーブルを参照することなくドット形成の有無を決定する場合は、大きくは2段階の操作を経ている。すなわち、先ず第1段階では、コード化された個数データを大中小の各種ドットの個数に変換する。続く第2段階では、順序値マトリックスに従って各画素についてのドット形成の有無を決定する。ここで、図38に示されているように、コード化された個数データと、大中小ドットについての個数の組合せとは、一対一の関係にある。換言すれば、コード化された個数データが1つ与えられれば、各種ドットについての個数の組合せは一意的に決



定することができる。

#### 【0227】

一方、画素群内に形成される各種ドットの個数が復号されると、図39に示したように、順序値マトリックスに従って、各画素についてのドット形成の有無が決定される。すなわち、順序値マトリックスが決まっていれば、各種ドットについての個数の組合せと、各画素についてのドット形成の有無とは、一対一の関係にある。上述したように、各種ドットについての個数の組合せは、コード化された個数データから一意的に決定されるから、結局、順序値マトリックスが決まっていれば、コード化された個数データから、画素群内の各画素について、各種ドットの形成有無は一意的に決定されることになる。

#### 【0228】

ここで、各種ドットの個数と順序値マトリックスとに従って、各画素についてのドット形成の有無を決定する操作を更に詳しく観察すると、次のようなことが分かる。すなわち、各種ドットの個数の組合せは、ある順序値の画素にはどのようなドットが形成されるかを決定しており、順序値マトリックスは、そのような順序値の画素が、画素群中のどの位置にあるかを決定している。例えば、図39に示した例では、大ドット1個、中ドット2個、小ドット1個であるから、順序値「1」の画素には大ドットが形成され、順序値「2」および「3」の画素には中ドットが形成され、順序値「4」の画素には小ドットが形成され、順序値「5」～「8」の画素にはいずれのドットも形成されないことがわかる。そして、これら順序値の画素が、画素群中ではどの位置にあるかについては、順序値マトリックスによって決定されている。このことから、コード化された個数データが与えられれば、ある順序値の画素にはどのようなドットが形成されるかについてまでは、一意的に決定することが可能である。

#### 【0229】

そこで、全ての個数データについて、順序値とその順序値を有する画素に形成されるドットの種類とを予め決定しておき、図40に示すような第2の変換テーブルに設定しておく。そして、画素群内の対象画素についてドット形成の有無を決定する場合には、順序値マトリックスを参照することで対象画素の順序値を求め、次いで第2の変換テーブルを参照することで、その順序値に形成されるドットの種類を取得すれば、ドット形成の有無を適切に決定することが可能となるのである。

#### 【0230】

こうして、1つの対象画素についてドット形成の有無を決定したら、選択した画素群内の全画素について、ドット形成の有無を決定したか否かを判断する（図23のステップS512相当）。未だ、ドット形成の有無を決定していない画素が残っていれば、ステップS506に相当する箇所まで戻って、新たな対象画素を選択し、続く一連の処理を行う。こうした処理を繰り返し、画素群内の全画素についてドット形成の有無を決定したと判断されたら、今度は、個数データが供給された全画素群について処理を終了したか否かを判断する（ステップS514相当）。そして、未処理の画素群が残っていれば、一番初めまで戻って新たな画素群を選択し、続く一連の処理を行う。こうした処理を繰り返し、全ての画素群について処理が終了したら、第2実施例のドット形成有無決定処理を終了する。

#### 【0231】

以上、第2実施例の画像印刷処理中で行われる個数データ生成処理、およびドット形成有無決定処理について詳しく説明した。上述した第2実施例の個数データ生成処理では、画素群についての分類番号と画素群階調値とを決定したら、第1の変換テーブルを参照するだけで、直ちにコード化された個数データを得ることができる。従って、コード化された個数データを極めて迅速に生成することができる。また、上述した第2実施例のドット形成有無決定処理では、コード化された状態の個数データを受け取ると、順序値マトリックスを参照して対象画素の順序値を取得した後、個数データと順序値とから第2の変換テーブルを参照することで、個数データを復号することなく各種ドットについての形成有無を決定することができる。このため、画素毎にドット形成有無を決定する処理を迅速に実行することが可能となり、延いては画像を迅速に出力することが可能となる。



## 【0232】

このように、個数データ生成処理およびドット形成有無決定処理は、いずれも極めて迅速に実行することが可能であるが、これに加えて、処理内容も極めて簡素なものとすることができる。以下では、この点につき、それぞれの処理について若干の補足説明を行う。

## 【0233】

初めに、個数データ生成処理について説明する。ディザ法を使用して、第1の変換テーブルを参照することなくドット個数を決定する場合、図29および図30に示されるような複雑な処理を行う必要がある。更に、得られたドット個数の組合せをコード化しなければならない。これに対して、第1の変換テーブルを参照すれば、同様の処理を、図35に示すような極めて簡素な処理で行うことができる。

## 【0234】

加えてディザ法を用いた方法では、図30に示されているように、画素群内の各画素について大中小ドットの形成有無を判断する際には、大ドットの形成有無、および中ドットの形成有無によって条件分岐が発生する。CPUでは高速な処理を可能とするために、パイプライン処理と呼ばれる技術が採用されているが、条件分岐が発生すると、パイプライン処理による効果を得ることができない。場合によっては、パイプライン処理を行うことで処理速度が低下してしまうことすらある。これに対して、第1の変換テーブルを参照する場合は、図35に示したように条件分岐を伴うことなく、コード化された個数データを得ることができ、従って、パイプライン処理の効果を十分に引き出して迅速な処理を行うことが可能である。第2実施例の個数データ生成処理は、単に処理が簡素なだけでなく、このような点からも高速処理に適した処理であると言える。

## 【0235】

次に、ドット形成有無を決定する処理について説明する。大中小各ドットについての形成の有無を、第2の変換テーブルを用いることなく決定する場合、図37に示す手順を踏むことになる。すなわち、初めに大ドットについて判断し、次いで中ドット、小ドットと、順番に判断していくことになる。これに対して、第2の変換テーブルを利用する場合は、テーブルを1回参照するだけという極めて簡単な操作で、大中小いずれのドットが形成されるかを決定することができる。

## 【0236】

加えて、大ドット、中ドット、小ドットと順番に判断する処理には、多くの条件分岐が存在する。すなわち、大ドットを形成すると判断された場合には、中ドットや小ドットについての判断は不要となる。また、中ドットを形成すると判断されると、小ドットのについての判断は不要となる。更に、小ドットも形成されないと判断された場合にだけ、その画素にはドットを形成しないものと判断される。このように、大ドット、中ドット、小ドットについての判断結果に応じて、以降の処理内容が異なっており、従って、多数の条件分岐が含まれている。これに対して、第2の変換テーブルを参照する場合は、こうした条件分岐を伴うことなく、大中小各ドットについての形成有無を直ちに決定することができる。このため、上述した個数データ生成処理と同様に、パイプライン処理の効果を十分に引き出して、一層迅速な処理を行うことが可能となる。上述した第2実施例の個数データ生成処理およびドット形成有無決定処理は、単に処理が簡素なだけでなく、このような点からも高速処理に適した処理であると言える。

## 【0237】

## C-4. メモリ使用量：

最後に、第2実施例の画像印刷処理におけるメモリ使用量について説明する。図41は、第1の変換テーブルを記憶するために要するメモリ量（データサイズ）を、各種条件下で試算した結果を示す説明図である。図27に示した第1実施例と同様に、第2実施例においても第1の変換テーブルのデータサイズは、分類番号の個数と、画素群階調値が取り得る範囲と、個数データ1つ当たりのデータ長とによって決定される。ここで、分類番号の個数は、ディザマトリックスのサイズと画素群の大きさによって決定される。また、個数データのデータ長は、画素群当たりの状態数、すなわち、1つの画素群内で起こり得

る大中小ドットの個数の組合せの種類によって決定される。例えば、前述したように、1つの画素群が8つの画素で構成されており、1つの画素が取り得る状態が、大ドットを形成、中ドットを形成、小ドットを形成、ドットを形成しないの4つの状態である場合は、大中小ドットの個数の組合せは165通り存在している。165通りであれば8ビットのデータ長があれば表現することができるから、個数データ1つ当たりのデータ長は1バイトとなる。同様に、1つの画素群が16個の画素から構成されている場合は、個数データ1つ当たりのデータ長は10ビット必要となるので、2バイトのデータとなる。

#### 【0238】

図41には、ディザマトリックスのサイズと、画素群の大きさとの組合せ毎に、変換テーブルのデータ量を試算した結果が示されている。尚、図41の右側には、個数データ1つ当たりに必要なデータ長が示されている。図41の試算結果に示されているように、第2実施例においても第1の変換テーブルのデータ量は、高々1Mバイトであり、実際には256Kバイト～512Kバイト程度に収まると考えられる。この程度であれば、一般的なコンピュータのキャッシュメモリにも十分に収まる程度の小さなデータ量であり、また、デジタルカメラ120などの画像機器やカラープリンタ200のメモリにも、十分に搭載可能である。

#### 【0239】

ここで、図41に表示されているデータ圧縮率について付言しておく。図41に示されたデータ圧縮率とは、画素毎にドット形成の有無を表すデータを出力する代わりに、画素群内に形成されるドットの個数の組合せをコード化した状態で出力することで、データ量がどの程度圧縮されるかを表す指標である。データ圧縮率が大きくなる程、コンピュータ100からカラープリンタ200にデータを迅速に出力することが可能となる。データ圧縮率は、次のようにして算出される。例えば、上述した実施例の場合について説明すると、画素群内に含まれる画素は8個である。また、画素単独で、大ドットを形成、中ドットを形成、小ドットを形成、ドットを形成せず、の4つの状態を取り得るから、1画素当たり2ビットのデータ長が必要となる。従って、画素毎に形成するドットの種類を表現しようとする、画素群当たり16ビットのデータ長が必要となる。これに対して、ドット個数の組合せをコード化してやれば、画素群当たり8ビットあれば足りる。結局、16ビットのデータ長を8ビットに圧縮することが可能となるので、データ圧縮率は0.5となる。

#### 【0240】

図41に示されているように、データ圧縮率は、条件によって変動するものの、おおむね0.5程度となっている。すなわち、大中小各ドットの個数の組合せをコード化することで、画素毎にドットの種類と形成有無とを出力する場合に比べて、データ量をほぼ半減させることができ、個数データをカラープリンタ200に迅速に出力することができる。この結果、画像を迅速に印刷することが可能となるのである。

#### 【0241】

次に、第2の変換テーブルを記憶するために必要なメモリ容量について説明する。図40に示したように、第2の変換テーブルには、順序値および個数データの組合せ毎にドット形成の有無が設定されている。画素群に含まれる画素数を8個とすれば、順序値は8通りの値を取る。また、形成可能なドットの種類が大ドット、中ドット、小ドットの3種類であるとすれば、個数データは165通りの値を取り得る。更に、ドット形成有無の判断結果は、大ドットを形成する、中ドットを形成する、小ドットを形成する、ドットを形成しないの4通りしか取り得ないから、1つの判断結果は2ビットあれば表現することができる。従って、図40に示した第2の変換テーブルは、 $8 \times 165 \times 2 = 2640$ ビット(0.322Kバイト)あれば記憶することができる。

#### 【0242】

図42は、第2の変換テーブルを記憶するために必要なメモリ量を、画素群の大きさ毎に試算した結果をまとめた説明図である。図示されているように、変換テーブルについても、数Kバイト程度あれば、十分記憶することができる。

## 【0243】

以上に説明したように、第2実施例の画像印刷処理においても、第1の変換テーブルや、第2の変換テーブル、複数の順序値マトリックスを記憶しておくために要するメモリ量は僅かである。加えて、上述した第2実施例の個数データ生成処理では、第1の変換テーブルを参照して直ちにコード化された個数データを得ており、図6に例示するようなディザマトリックスや、階調データを各種ドットの密度データに変換するための図31に例示するようなドット密度変換テーブルを参照する必要がない。このため、第2実施例においては、ディザマトリックスやドット密度変換テーブルを記憶しておく必要が無くなる。こうしたことを考慮すれば、第2実施例の画像印刷処理についても、メモリ容量が製品への搭載の障害となることはない。

## 【0244】

以上、各種の実施例について説明してきたが、本発明は上記すべての実施例に限られるものではなく、その要旨を逸脱しない範囲において種々の態様で実施することができる。例えば、以上の実施例では、印刷用紙上にドットを形成して画像を印刷する場合について説明したが、本発明の適用範囲は画像を印刷する場合に限られるものではない。例えば、液晶表示画面上で輝点を適切な密度で分散させることにより、階調が連続的に変化する画像を表現する液晶表示装置などにも、本発明を好適に適用することができる。

## 【図面の簡単な説明】

## 【0245】

【図1】印刷システムを例にとって本発明の概要を説明するための説明図である。

【図2】本実施例の画像処理装置としてのコンピュータの構成を示す説明図である。

【図3】本実施例のカラープリンタの概略構成を示す説明図である。

【図4】インク吐出用ヘッドにおけるインクジェットノズルの配列を示す説明図である。

【図5】第1実施例の画像印刷処理の全体的な流れを示すフローチャートである。

【図6】ディザマトリックスの一部を概念的に例示した説明図である。

【図7】ディザマトリックスを参照しながら各画素についてのドット形成の有無を判断している様子を概念的に示した説明図である。

【図8】ディザ法を用いて画像データをドット形成の有無を表すデータに変換している様子を示した説明図である。

【図9】ドット個数を示すデータから画素毎にドット形成の有無を表すデータを生成する様子を示した説明図である。

【図10】ブルーノイズマスク特性を有するディザマトリックス、およびグリーンノイズマスク特性を有するディザマトリックスに設定されている閾値の空間周波数特性を概念的に例示した説明図である。

【図11】第1実施例の個数データ生成処理の流れを示すフローチャートである。

【図12】画素群毎の分類番号を決定するための考え方を示した説明図である。

【図13】画素群の分類番号を決定する方法を示した説明図である。

【図14】着目画素群の分類番号を決定する方法を具体的に示した説明図である。

【図15】画素群の分類番号と画素群階調値とから個数データを取得するために参照される第1の変換テーブルを概念的に示した説明図である。

【図16】画素群の分類番号と画素群階調値との組合せに応じて適切な個数データを決定する様子を概念的に示した説明図である。

【図17】第1実施例のドット形成有無決定処理の流れを示すフローチャートである。

【図18】第1実施例のドット形成有無決定処理において各画素についてのドット形成の有無が決定される様子を概念的に示した説明図である。

【図19】対象画素についてのドット形成の有無を決定するために参照される第2の変換テーブルを概念的に示した説明図である。

【図20】第1の変形例における画像印刷処理の流れを示したフローチャートである。

。

【図 2 1】解像度調整処理の概要を示した説明図である。

【図 2 2】第 1 の変形例における個数データ生成処理の流れを示すフローチャートである。

【図 2 3】第 2 の変形例におけるドット形成有無決定処理の流れを示すフローチャートである。

【図 2 4】第 2 の変形例のドット形成有無決定処理中で参照される複数の順序値マトリックスを生成する方法について示した説明図である。

【図 2 5】画素群に対応する順序値マトリックスを選択する方法を示した説明図である。

【図 2 6】画素群の座標値から適用する順序値マトリックスを選択する方法を具体的に示した説明図である。

【図 2 7】第 1 の変換テーブルを記憶するためのメモリ使用量を試算した結果を示す説明図である。

【図 2 8】順序値マトリックスを記憶するために必要なメモリ量を試算した結果を示す説明図である。

【図 2 9】画素群内に形成される大ドット、中ドット、小ドットの個数をディザ法により決定する処理の流れを示すフローチャートである。

【図 3 0】大ドット、中ドット、小ドットの形成有無を判断するハーフトーン処理の流れを示すフローチャートである。

【図 3 1】画像データの階調値を大中小各ドットについての密度データに変換するためのドット密度変換テーブルを概念的に示した説明図である。

【図 3 2】ディザ法を適用しながら大中小各ドットの形成有無を判断している様子を概念的に示した説明図である。

【図 3 3】画素群毎に大ドット、中ドット、小ドットの形成個数が得られた様子を概念的に示した説明図である。

【図 3 4】画素群に形成される大中小各ドットの個数の組合せとコードデータとが対応付けて設定された対応テーブルを示す説明図である。

【図 3 5】第 2 実施例の個数データ生成処理の流れを示すフローチャートである。

【図 3 6】第 2 実施例の個数データ生成処理において参照される第 1 の変換テーブルを概念的に示した説明図である。

【図 3 7】第 2 の変換テーブルを参照することなく大中小各ドットの形成有無を決定する処理の流れを示すフローチャートである。

【図 3 8】コード化された個数データを復号するための復号テーブルを概念的に示した説明図である。

【図 3 9】順序値マトリックスを参照しながら大中小の各種ドットについての形成有無を決定する様子を概念的に示した説明図である。

【図 4 0】第 2 実施例のドット形成有無決定処理で参照される第 2 の変換テーブルを概念的に示した説明図である。

【図 4 1】第 1 の変換テーブルを記憶するためのメモリ量を試算した結果を示す説明図である。

【図 4 2】第 2 の変換テーブルを記憶するためのメモリ量を試算した結果を示す説明図である。

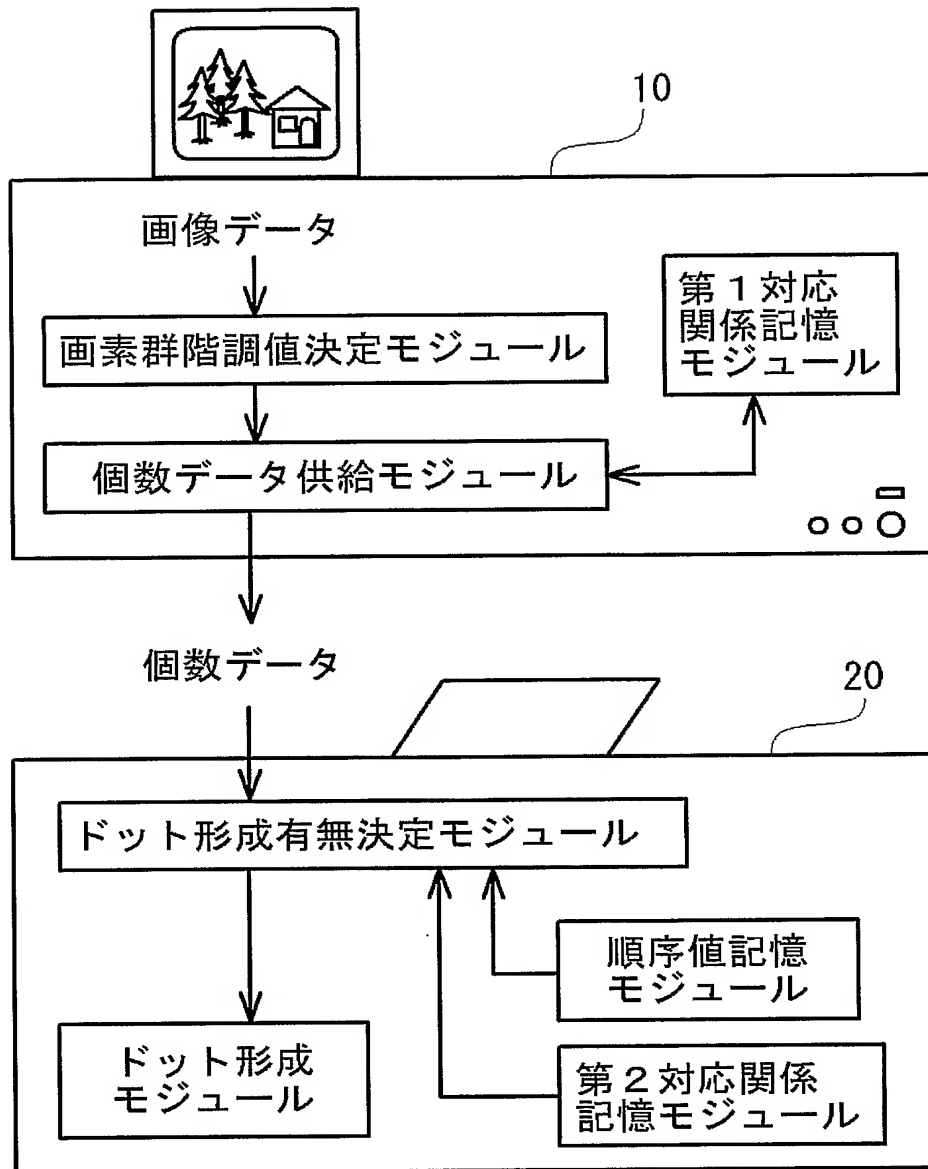
#### 【符号の説明】

##### 【0246】

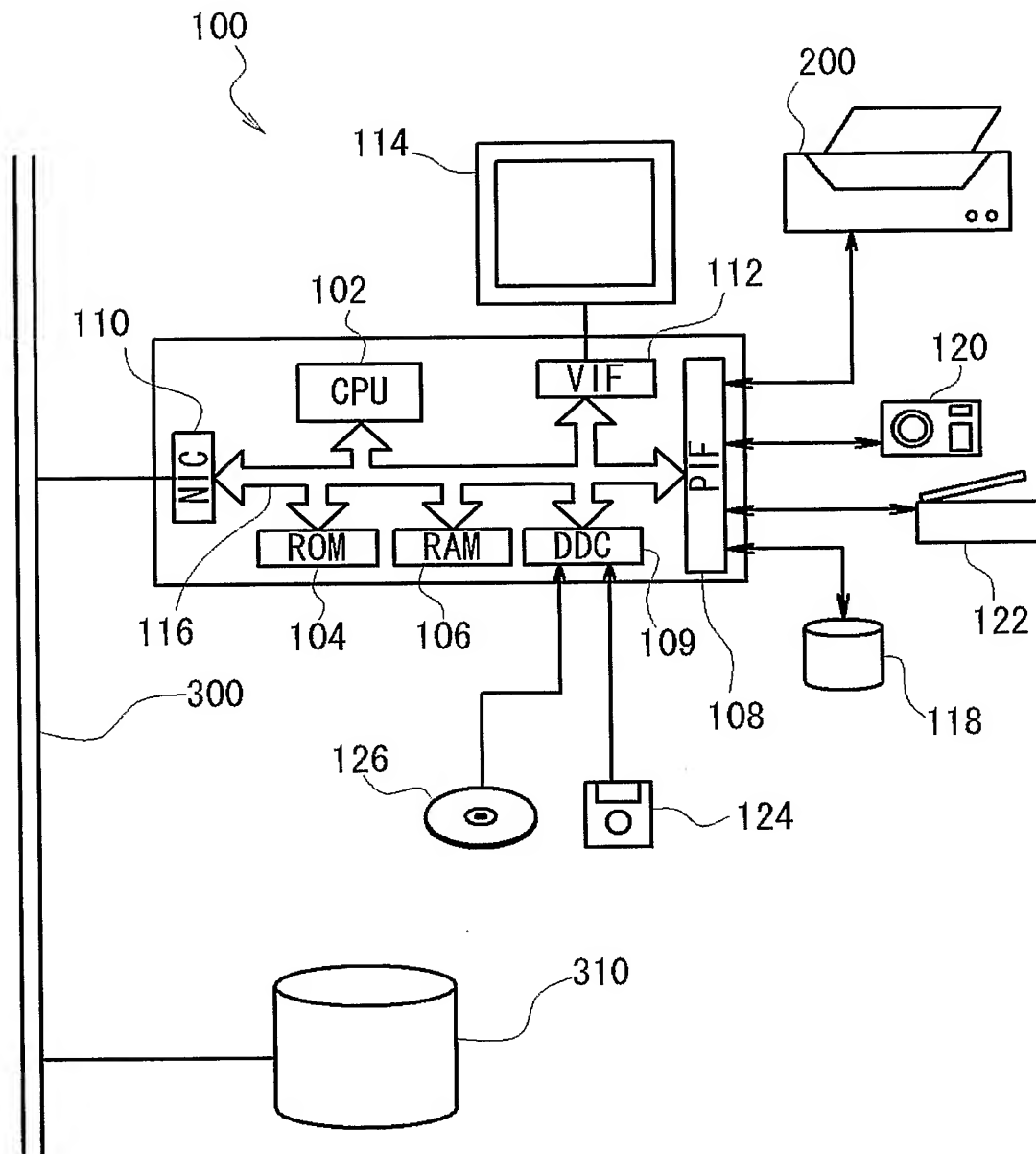
- 10…コンピュータ、
- 20…カラープリンタ、
- 100…コンピュータ、
- 108…周辺機器インターフェースPIF、
- 109…ディスクコントローラDDC、
- 110…ネットワークインターフェースカードNIC、
- 112…ビデオインターフェースVIF、

1 1 6 …バス、 1 1 8 …ハードディスク、 1 2 0 …デジタルカメラ、  
1 2 2 …カラスキャナ、 1 2 4 …フレキシブルディスク、  
1 2 6 …コンパクトディスク、 2 3 0 …キャリッジモータ、 2 3 5 …モータ、  
2 3 6 …プラテン、 2 4 0 …キャリッジ、 2 4 1 …印字ヘッド、  
2 4 2 …インクカートリッジ、 2 4 3 …インクカートリッジ、  
2 4 4 …インク吐出用ヘッド、 2 6 0 …制御回路、 2 0 0 …カラープリンタ、  
3 0 0 …通信回線、 3 1 0 …記憶装置

【書類名】 図面  
【図 1】

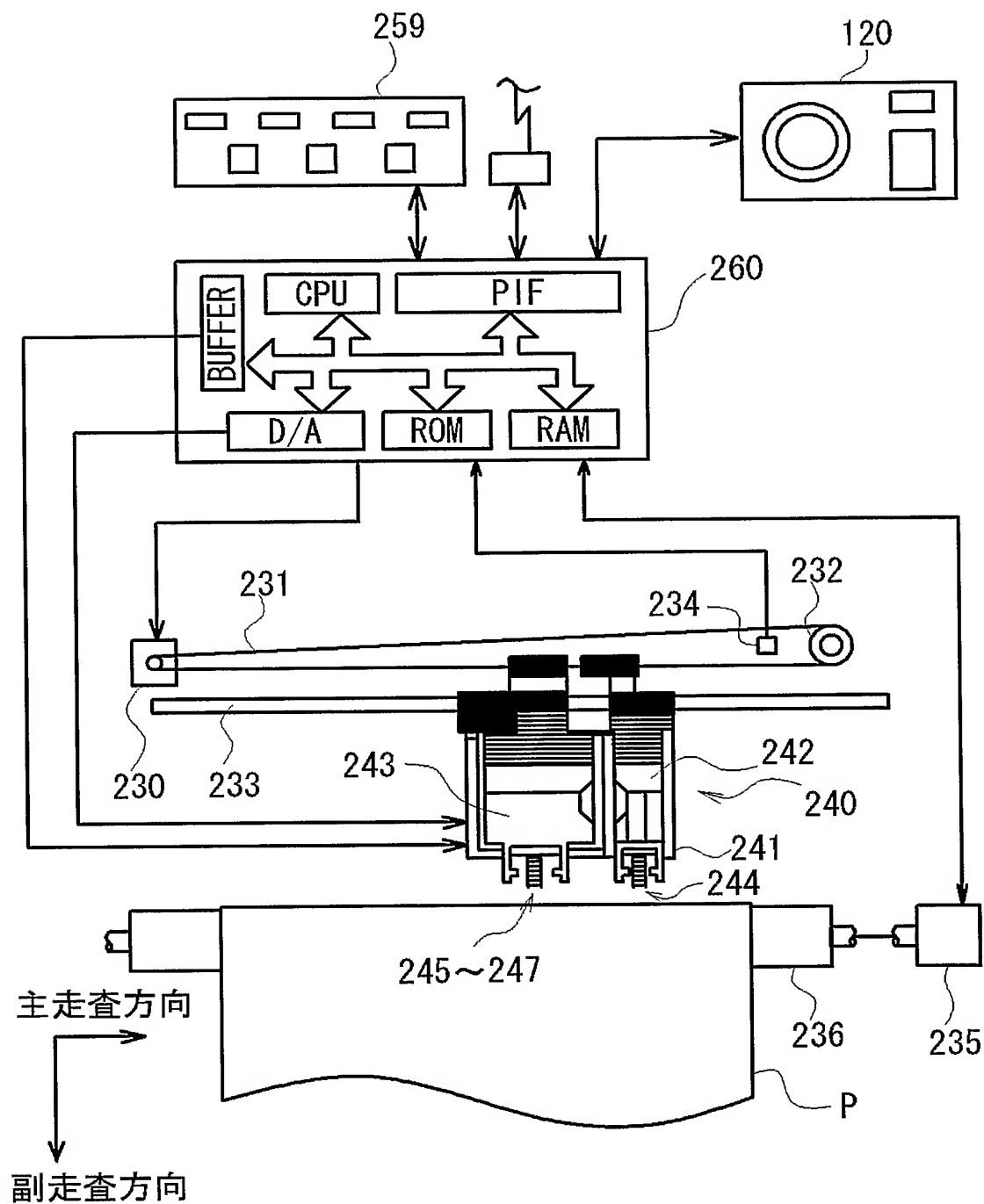


【図 2】

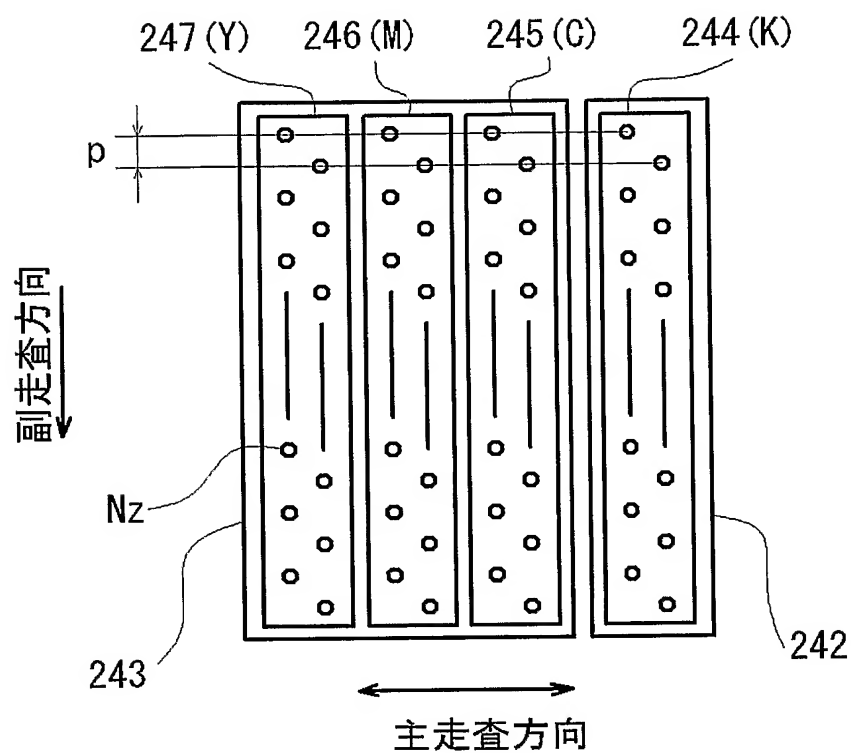




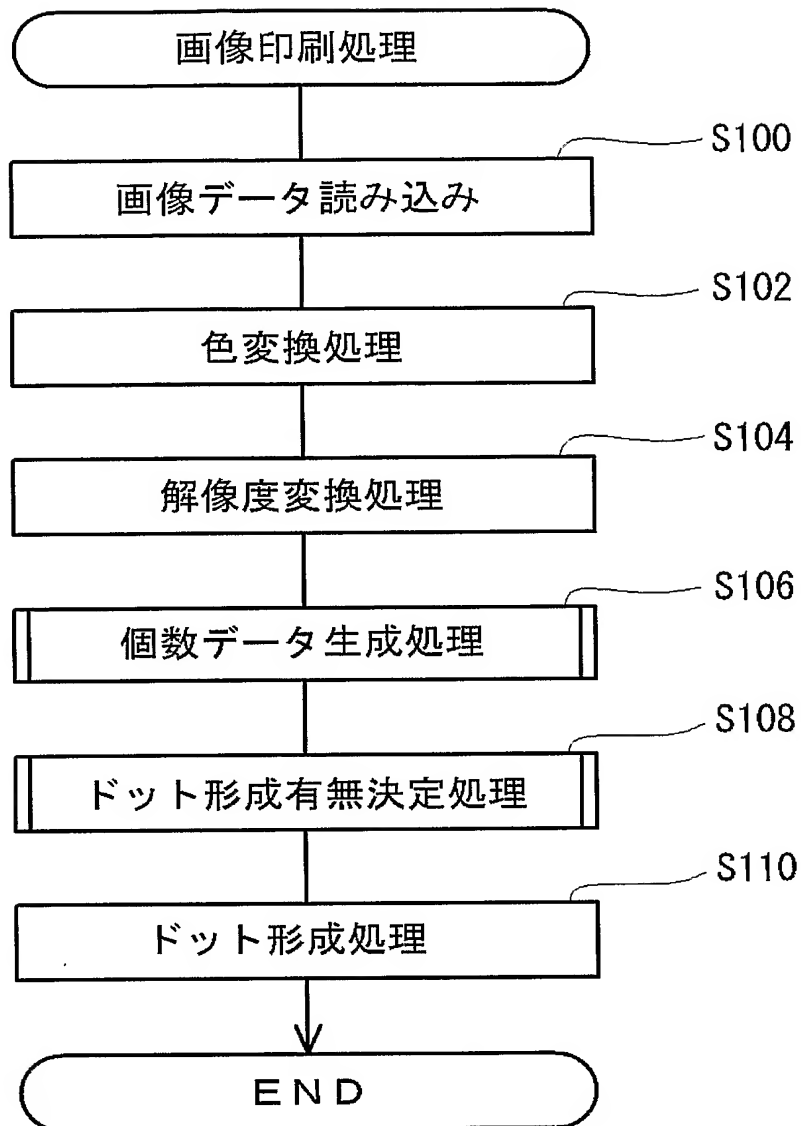
【図 3】



【図 4】



【図 5】



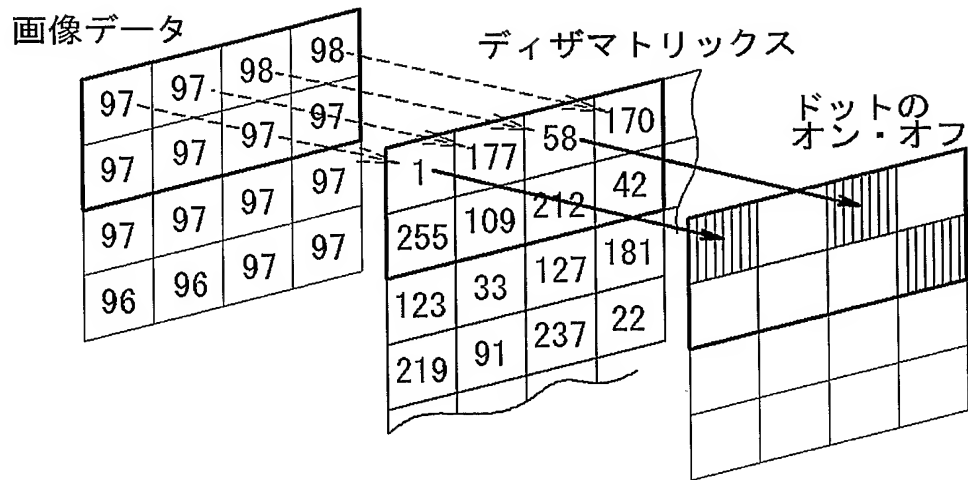
【図 6】

→ 128画素

↓ 64画素

1	177	58	170	70	186	79	161	94	171	16	164	24	158	227	6	133	157
255	109	212	42	242	5	223	48	209	67	248	81	234	132	56	120	253	
123	33	127	181	83	155	105	140	30	150	119	182	40	89	220	163	44	
219	91	237	22	121	198	61	188	111	215	3	98	201	140	10	95		
11	195	53	136	227	37	247	12	233	52	192	135	32	246	113	194		
99	144	107	184	68	172	97	151	77	173	84	237	123	61	167	46		
225	40	251	6	217	116	28	196	125	35	207	17	153	203	24			
87	169	78	162	59	146	211	64	254	142	72	178	87	118	228			
190	15	202	111	238	19	93	169	8	110	221	49	249	2	144			
74	246	134	43	174	128	230	50	216	154	26	168	79	184				
176	30	98	219	86	34	139	195	101	56	241	127	213	37				
69	148	196	2	159	247	89	11	136	185	92	14	108					
187	41	126	226	106	57	190	115	235	36	208	121	229					
81	214	92	53	145	204	27	166	74	157	82	165	31					
232	21	170	240	13	132	252	51	222	4	245	48						
130	155	55	115	183	78	122	196	102	180	65	173						
72	252	100	211	45	231	20	148	39	133	205							
202	32	179	5	163	95	191	86	239	111	29							
104	143	58	243	70	218	63	161	9	209								
64	236	129	110	151	25	199	49	172	68								
171	17	208	38	225	131	99	254	113									
80	191	87	174	76	189	13	143										
247	35	152	2	248	55												
140	108	197	127	96													
13	215	46	229														
164	90	180															
52	253																
205																	

【図 7】



【図 8】

(a)

97	97	98	98	98	98	98	98
97	97	97	97	97	97	98	98
97	97	97	97	97	97	97	97
96	96	97	97	97	97	97	97

(b)

1	177	58	170	70	186	79	161
255	109	212	42	242	5	223	48
123	33	127	181	83	155	105	140
219	91	237	22	121	198	61	188

(c)


(d)

3	4
3	2

【図 9】

(a)

3	4
3	2

(b)

1	177	58	170	70	186	79	161
255	109	212	42	242	5	223	48
123	33	127	181	83	155	105	140
219	91	237	22	121	198	61	188

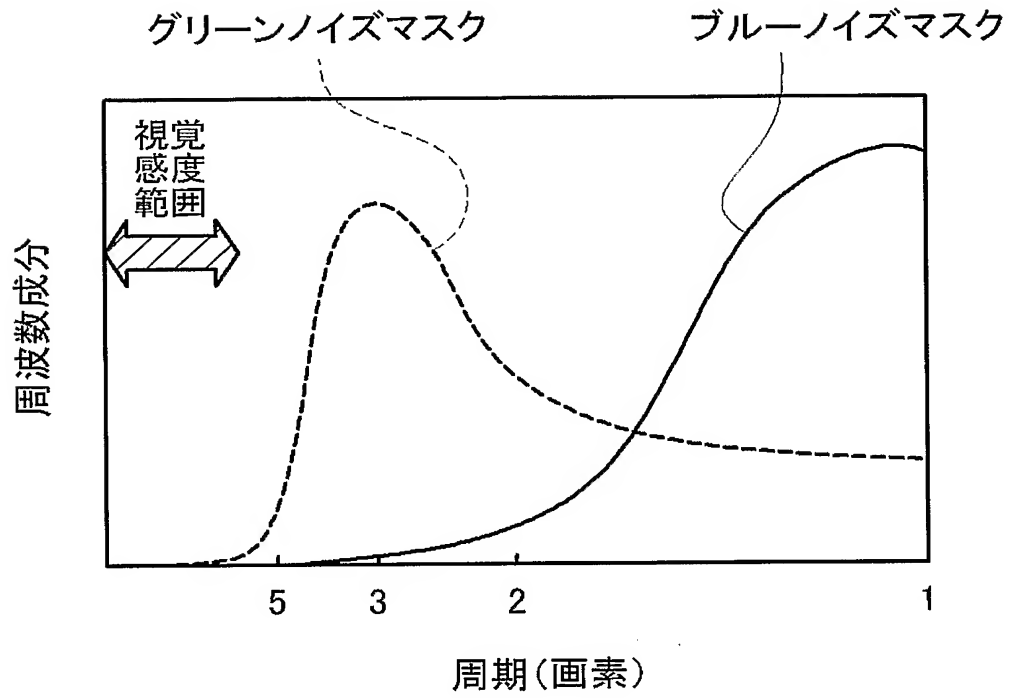
(c)


(d)

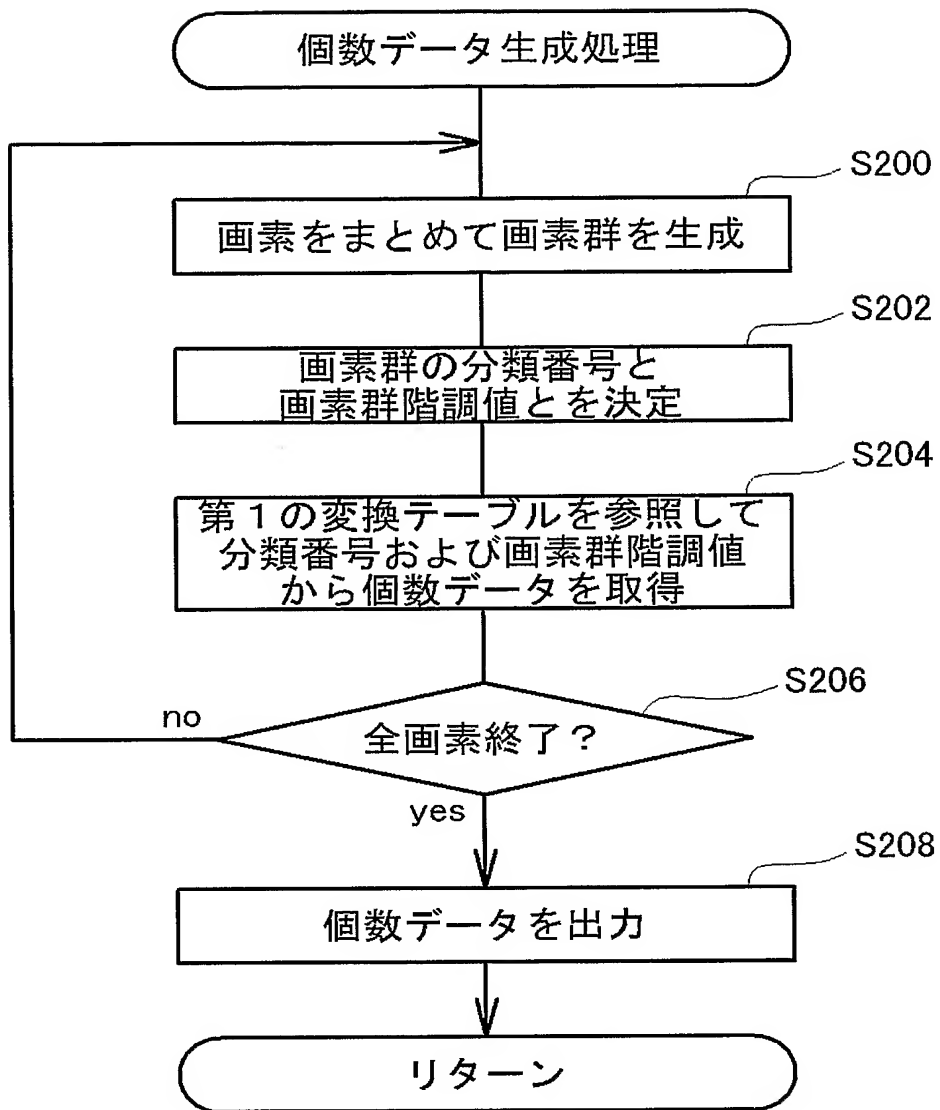
1	6	3	5	3	6	4	5
8	4	7	2	8	1	7	2
4	2	5	6	2	6	3	5
7	3	8	1	4	8	1	7



【図 10】

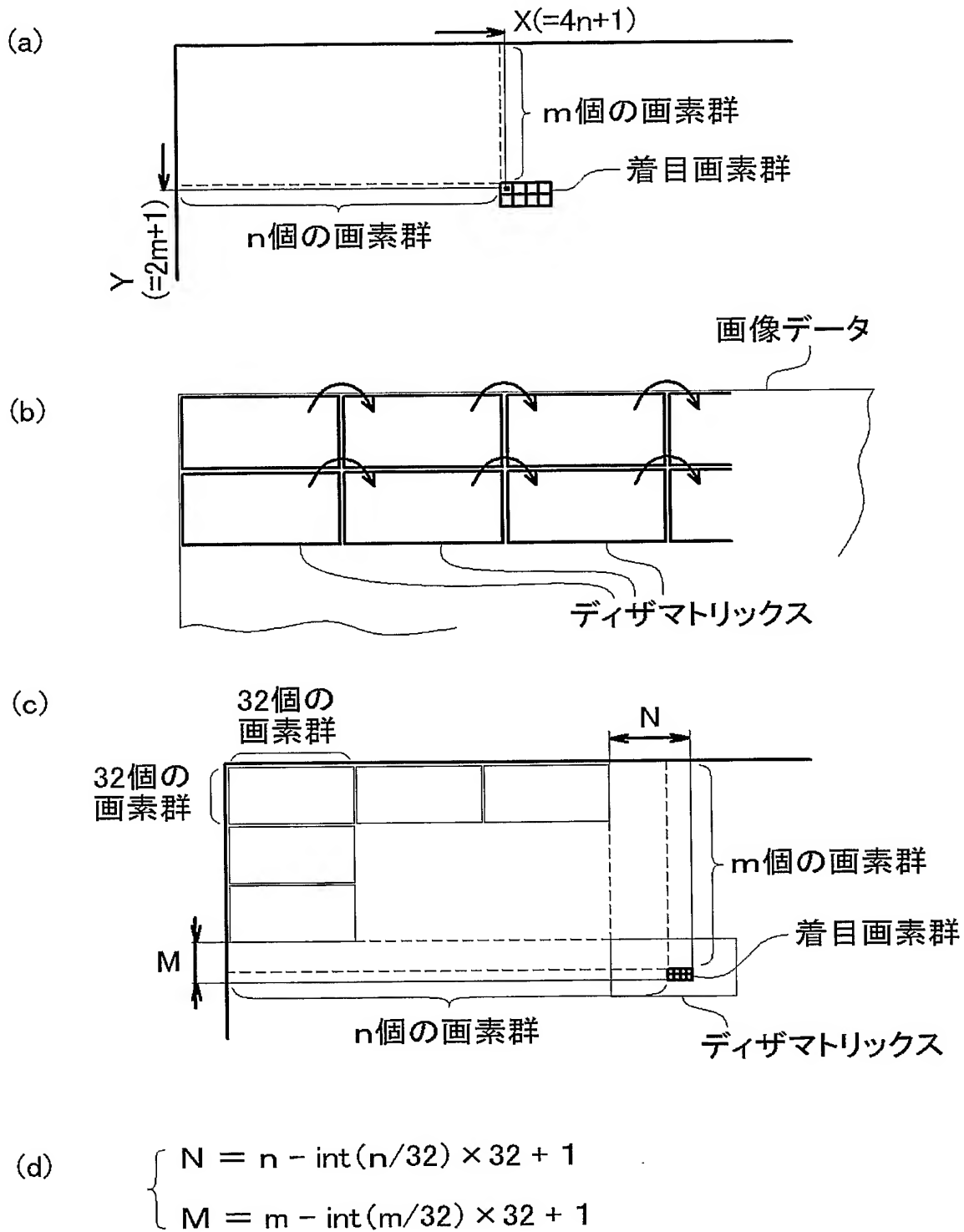


【図 11】

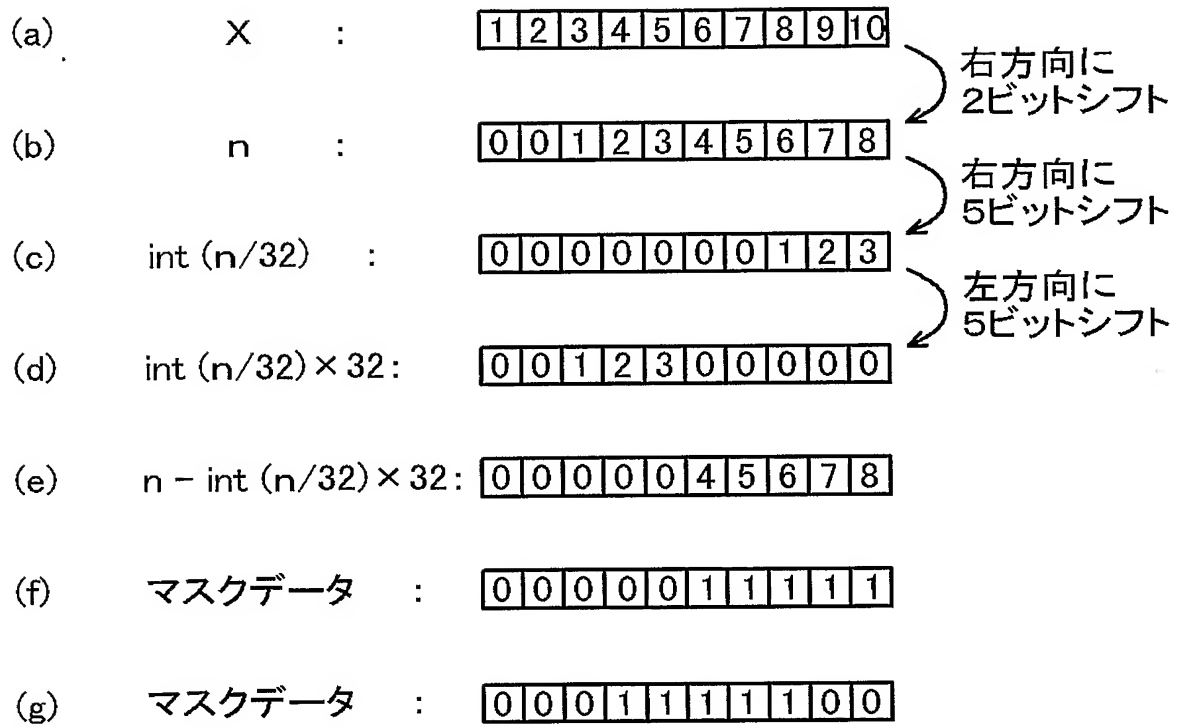




【図 13】



【図 1 4】



【図 15】

		画素群階調値								
		0	1	2	3	4	5	253	254	255
分類番号	1	0	1	1	1	1	1	7	7	8
	2	0	0	0	0	0	1	8	8	8
	3	0	0	0	0	0		8	8	8
	4	0	0	0	0				8	8
	5	0	0	0	0				8	8
	6	0	0	0					8	8
	7	0	0							
	8	0								
	9	0								
	10									

.....

↓

1024番まで

【図 16】

(a) ブロック1

1	177	58	170
255	109	212	42

(b) 画素群階調値:0


個数データ:0

(c) 画素群階調値:1

●			

個数データ:1

(d) 画素群階調値:2

●			

個数データ:1

(e) 画素群階調値:100

●		●	
			●

個数データ:3

(f) 画素群階調値:200

●	●	●	●
	●		●

個数データ:6

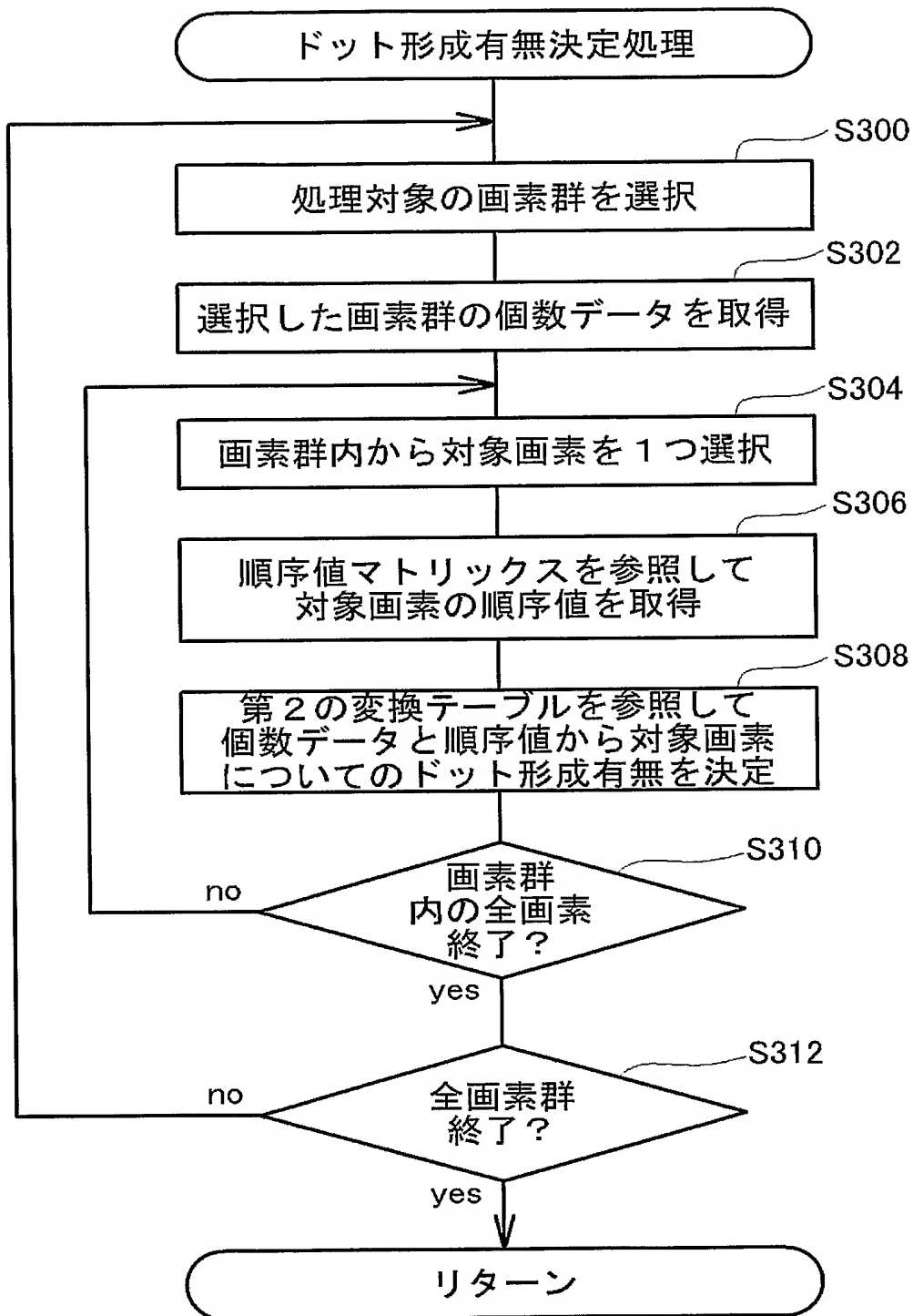
(g) 画素群階調値:255

●	●	●	●
●	●	●	●

個数データ:8

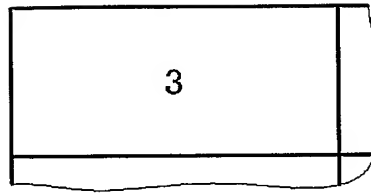


【図 17】



【図 18】

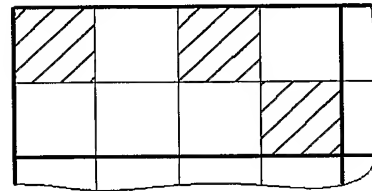
(a)



(b)

1	6	3	5
8	4	7	2

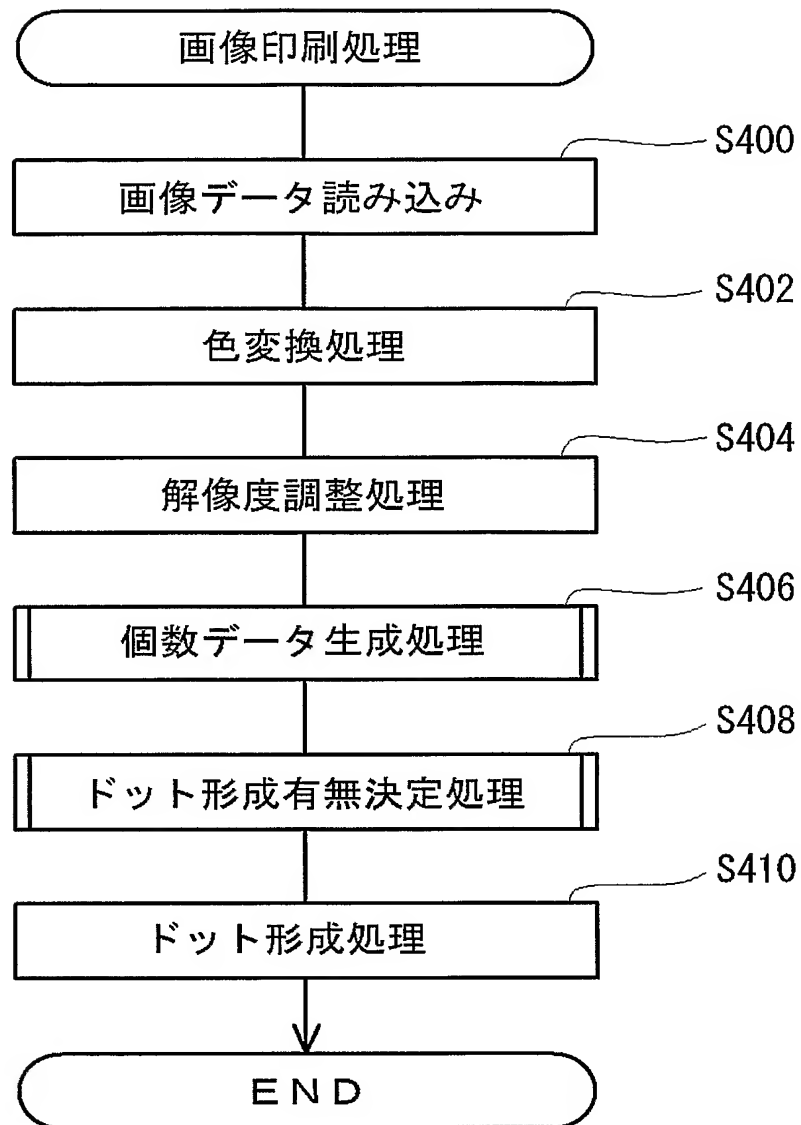
(c)



【図 19】

		個数データ								
		0	1	2	3	4	5	6	7	8
順序値	1	0	1	1	1	1	1	1	1	1
	2	0	0	1	1	1	1	1	1	1
	3	0	0	0	1	1	1	1	1	1
	4	0	0	0	0	1	1	1	1	1
	5	0	0	0	0	0	1	1	1	1
	6	0	0	0	0	0	0	1	1	1
	7	0	0	0	0	0	0	0	1	1
	8	0	0	0	0	0	0	0	0	1

【図 20】



【図 21】

(a)

97	98	98	98	
97	97	97	98	
96	97	97	97	

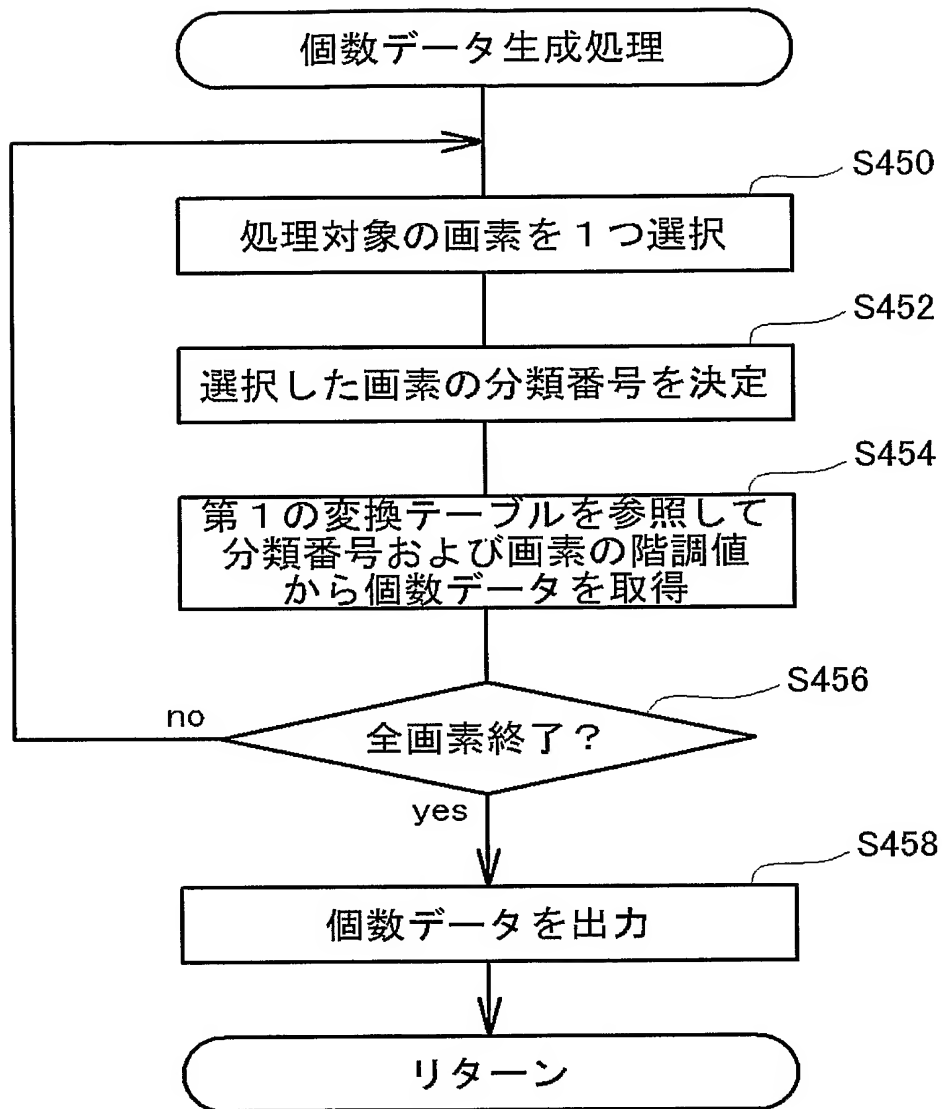
(b)

97	98	98	
97	97	98	
96	97	97	

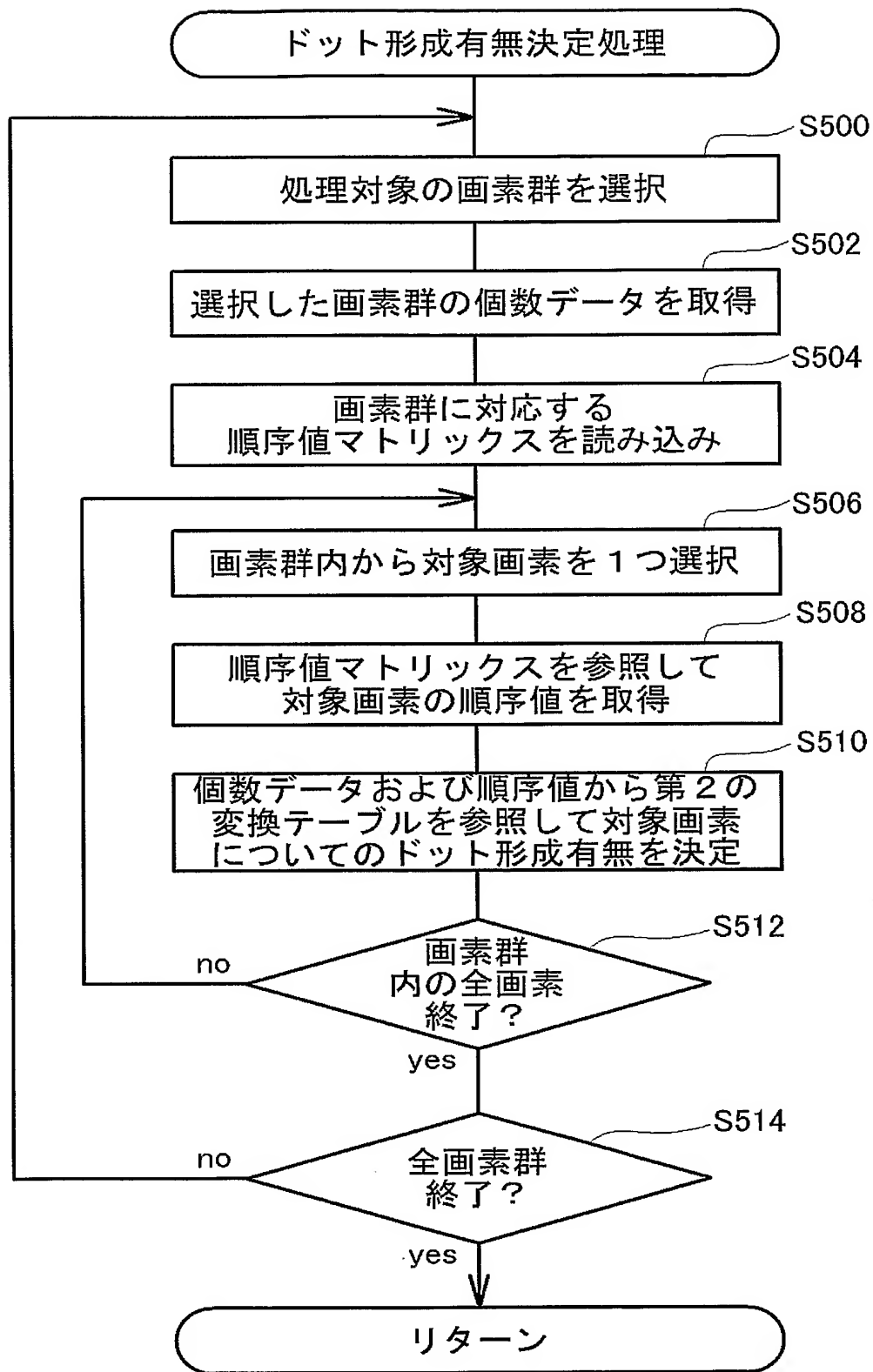
(c)

97	97	97	97	98	98	98	98	98	98	98	98	
97	97	97	97	98	98	98	98	98	98	98	98	
97	97	97	97	97	97	97	97	98	98	98	98	
97	97	97	97	97	97	97	97	98	98	98	98	
96	96	96	96	97	97	97	97	97	97	97	97	
96	96	96	96	97	97	97	97	97	97	97	97	

【図 22】

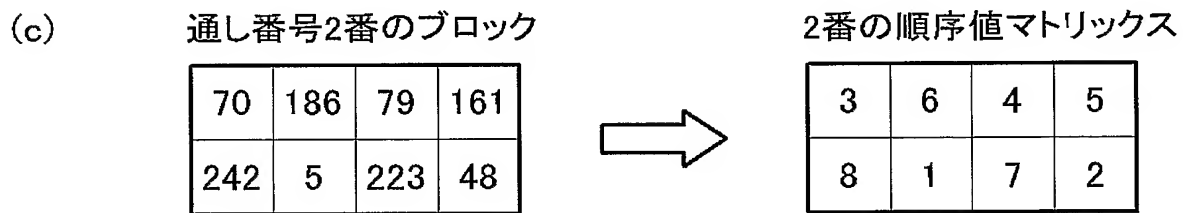
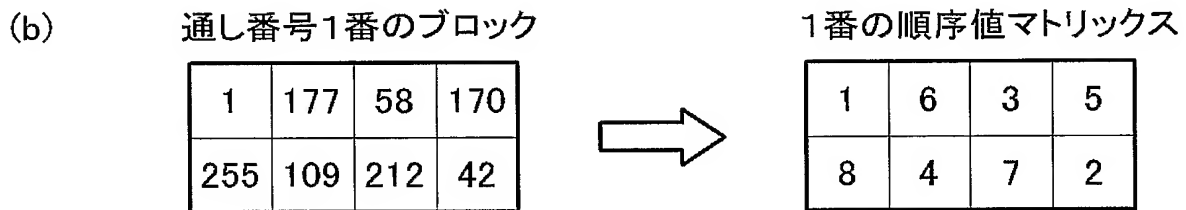
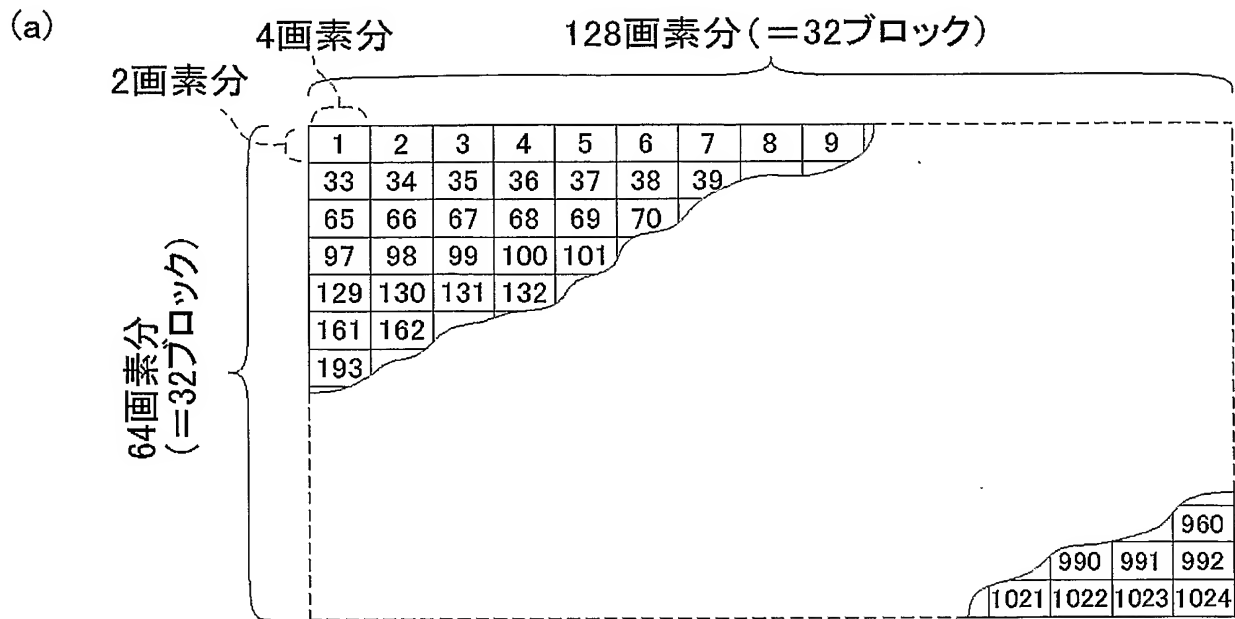


【図 23】

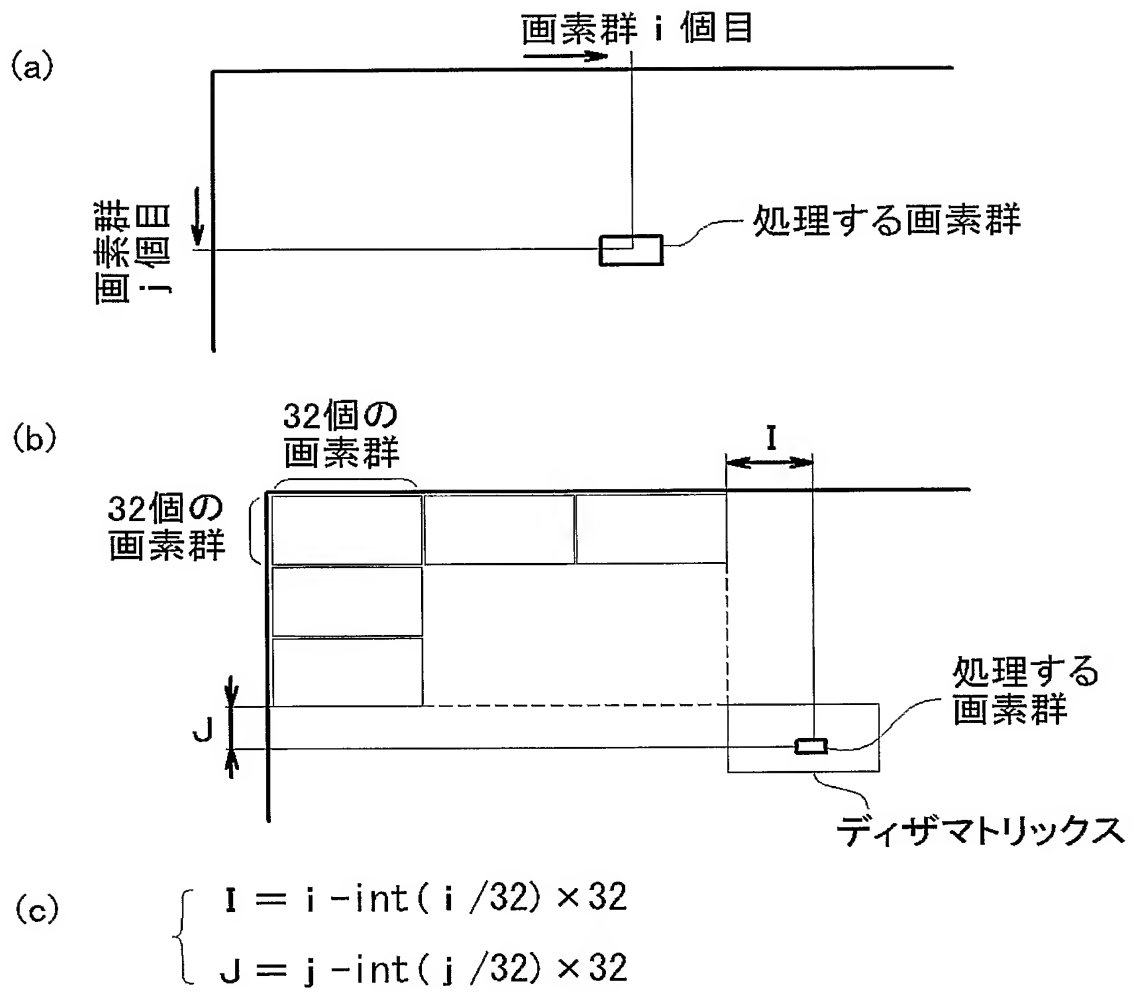




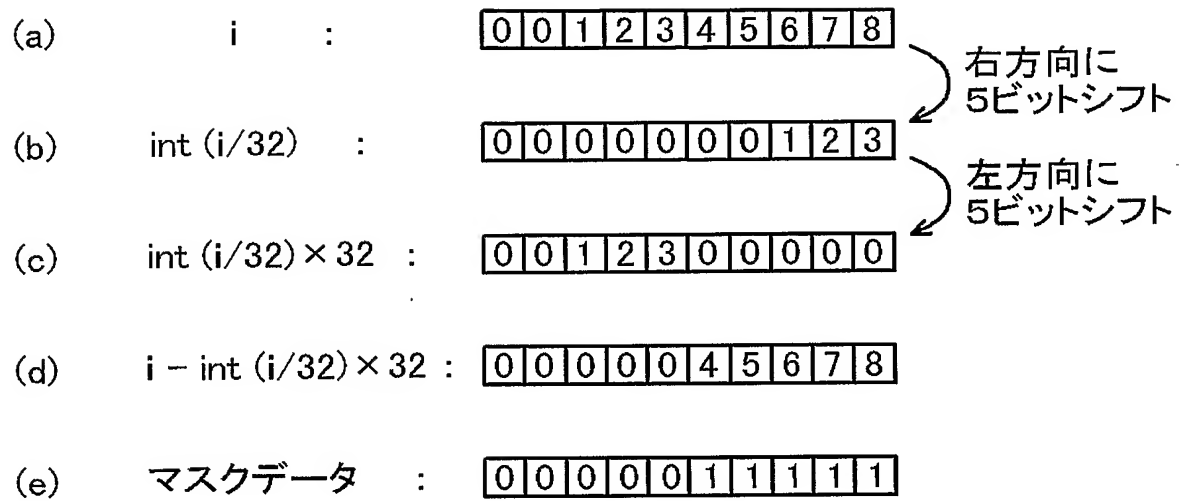
【図 24】



【図 25】



【図 26】



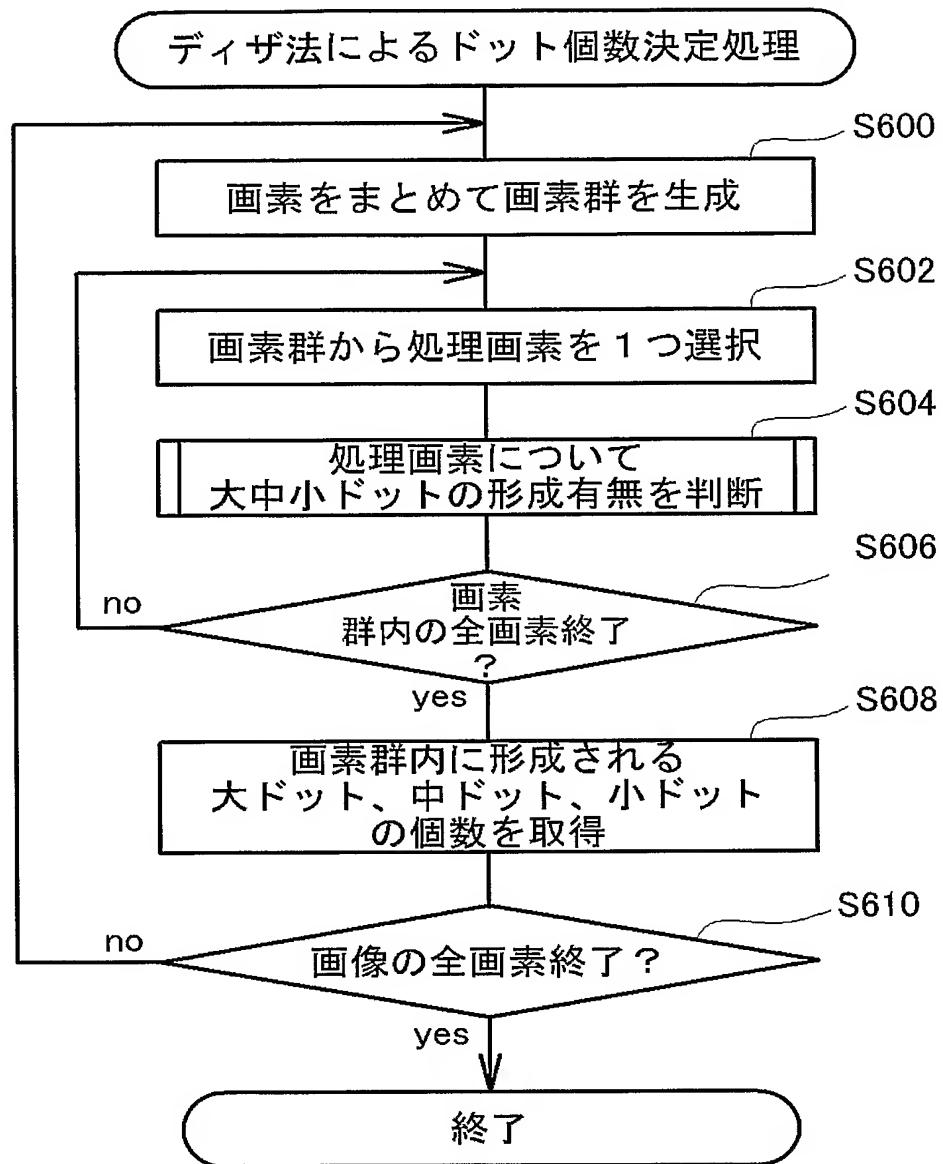
【図 27】

ディザマトリックス サイズ (画素)		画素群サイズ (画素)		ブロック 数	変換 テーブル サイズ (バイト)	状態数 ／ 画素群	使用 ビット 数
主走査 方向	副走査 方向	主走査 方向	副走査 方向				
64	64	2	2	1024	256K	5	3
		4	2	512	128K	9	4
		4	4	256	64K	17	5
128	64	2	2	2048	512K	5	3
		4	2	1024	256K	9	4
		4	4	512	128K	17	5
128	128	2	2	4096	1024K	5	3
		4	2	2048	512K	9	4
		4	4	1024	256K	17	5

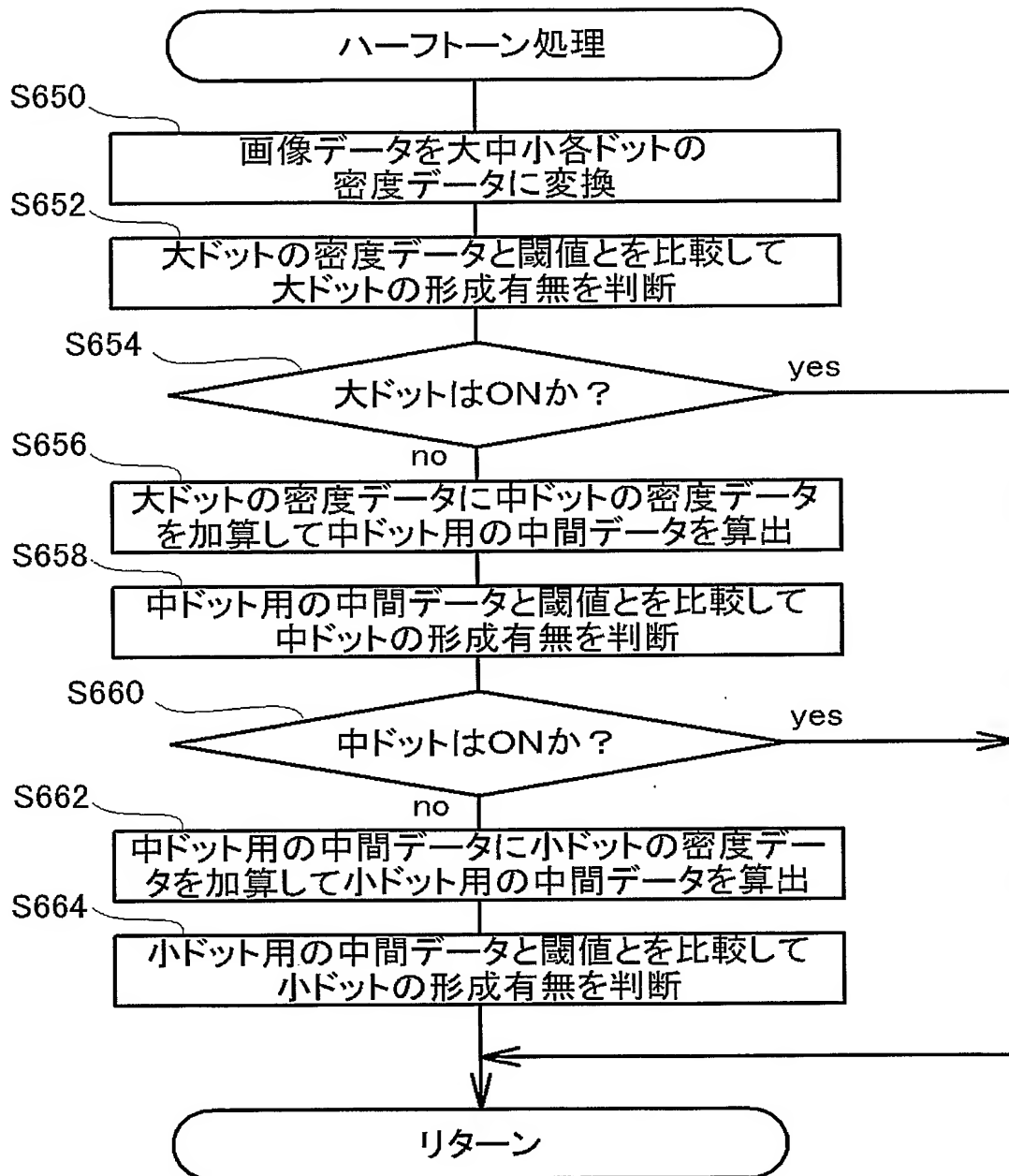
【図 28】

ディザマトリックス サイズ (画素)		画素群サイズ (画素)		順序値 マトリックス の個数	順序値 マトリックス のサイズ (バイト)	必要な メモリ量 (Kバイト)
主走査 方向	副走査 方向	主走査 方向	副走査 方向			
64	64	2	2	1024	1	1
		4	2	512	3	1.5
		4	4	256	8	2
128	64	2	2	2048	1	2
		4	2	1024	3	3
		4	4	512	8	4
128	128	2	2	4096	1	4
		4	2	2048	3	6
		4	4	1024	8	8

【図 29】

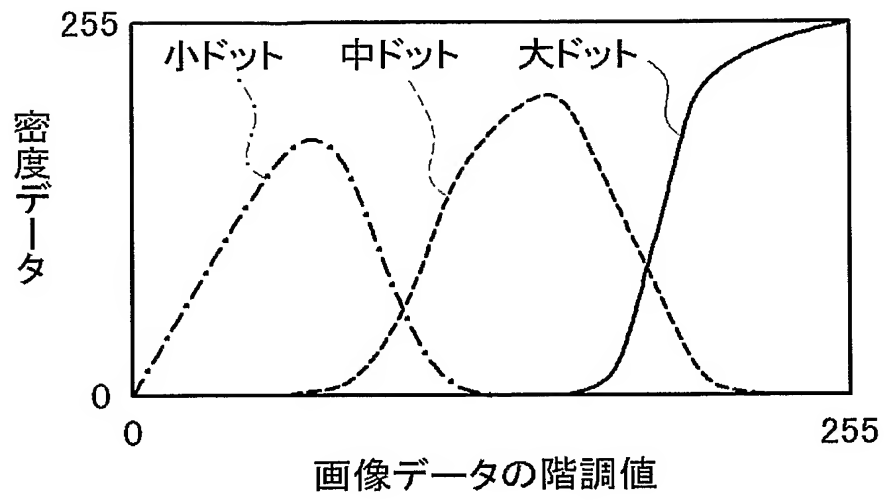


【図 30】

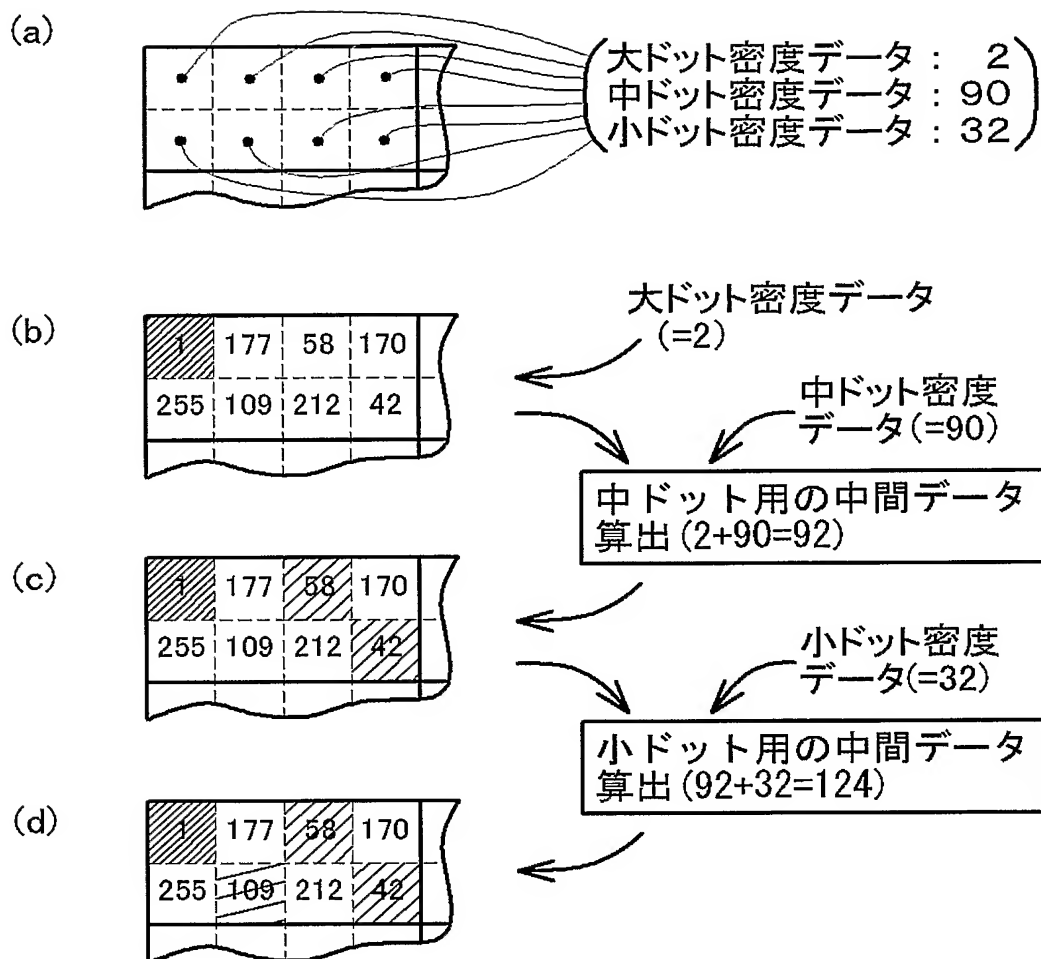




【図 31】



【図 3 2】



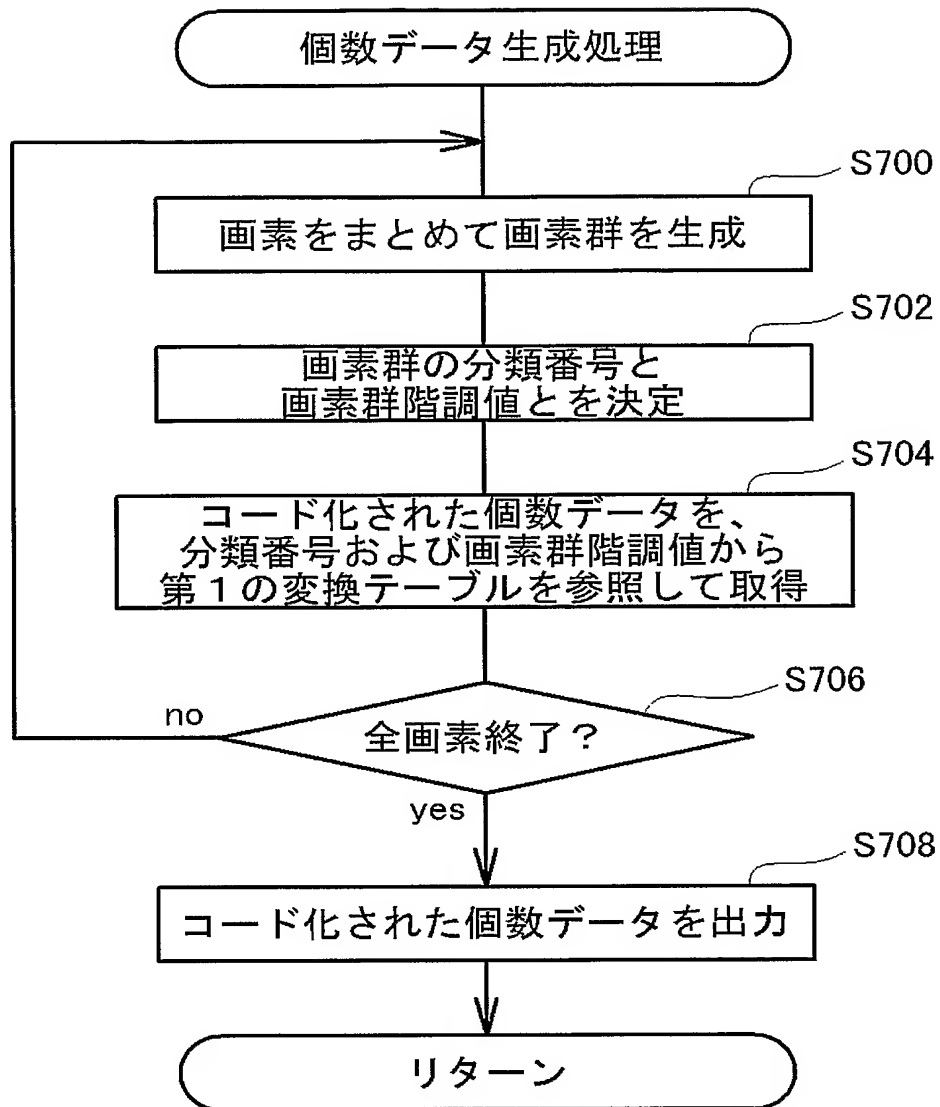
【図 33】

(大,中,小) =(1,2,1)	(0,4,0)	(0,3,1)	
(0,3,0)	(0,2,1)	(1,1,0)	
(0,2,3)	(1,2,1)	(0,3,0)	

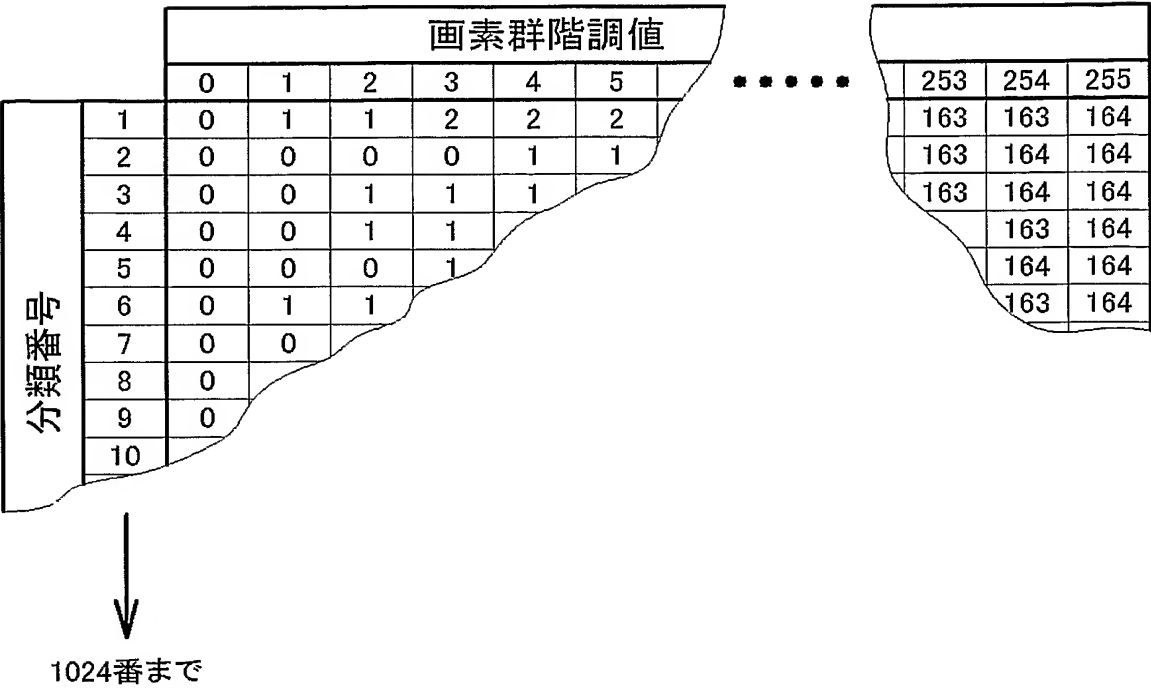
【図 34】

ドット個数			コード化 された 個数データ
大	中	小	
0	0	0	0
0	0	1	1
0	0	2	2
0	0	3	3
6	2	0	160
7	0	0	161
7	0	1	162
7	1	0	163
8	0	0	164

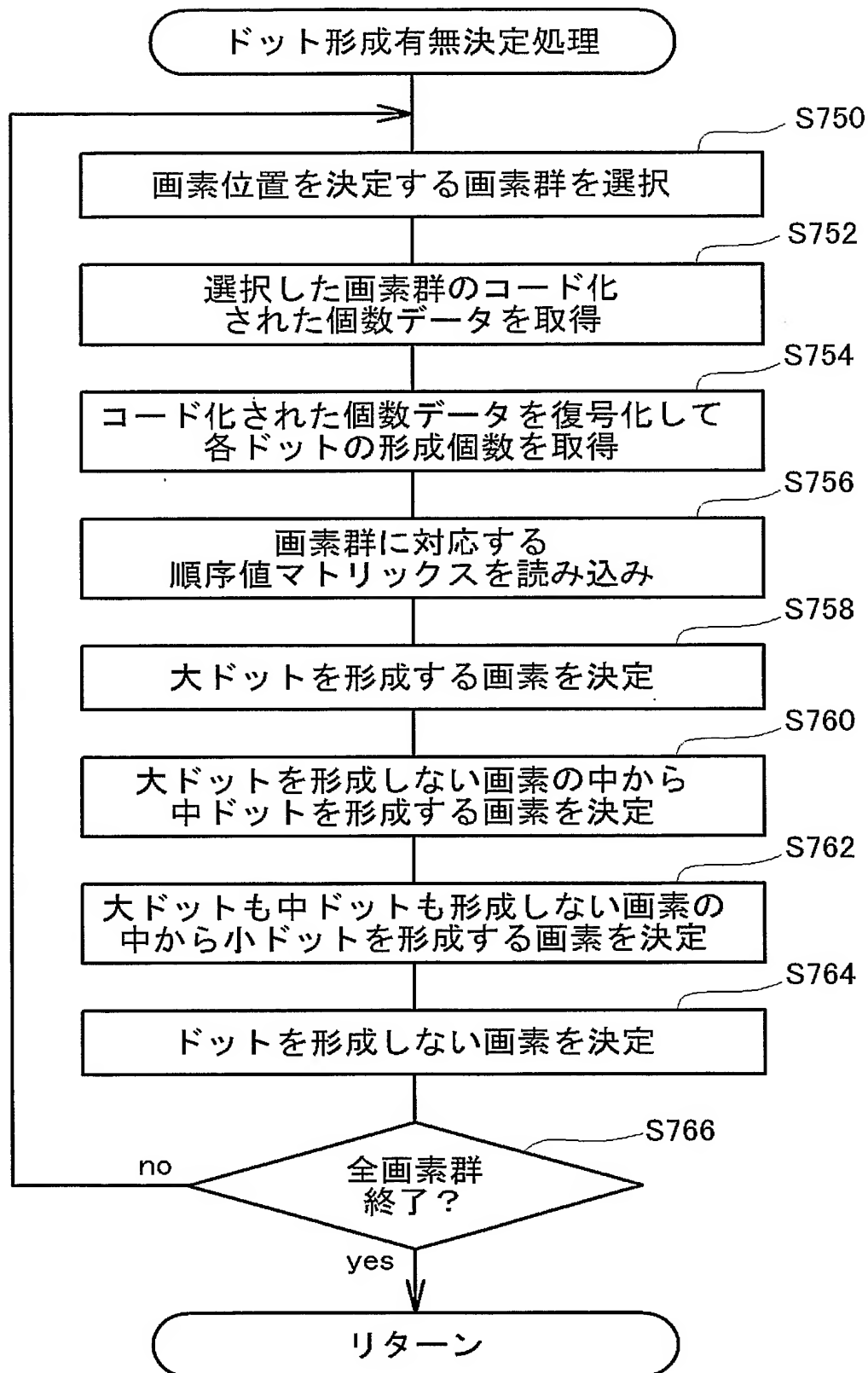
【図 35】



【図 3 6】



【図 37】

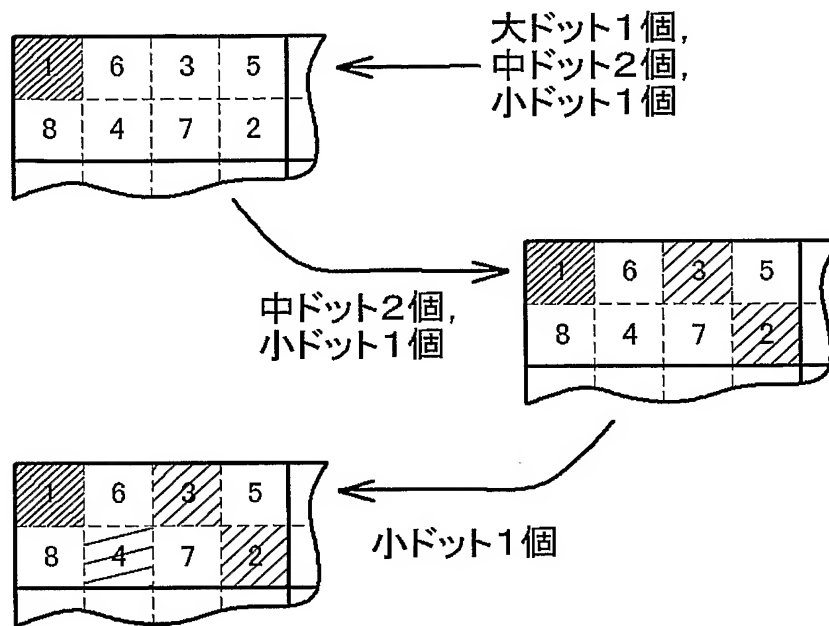




【図 38】

コード化 された 個数データ	ドット個数		
	大	中	小
0	0	0	0
1	0	0	1
2	0	0	2
3	0	0	3
160	6	2	0
161	7	0	0
162	7	0	1
163	7	1	0
164	8	0	0

【図 39】



【図 40】

		個数データ										
		0	1	2	3	4	5	6	7	162	163	164
順序値	1	0	1	1	1	1	1	1	1	3	3	3
	2	0	0	1	1	1	1	1	1		3	3
	3	0	0	0	1	1	1				3	3
	4	0	0	0	0	1	1					3
	5	0	0	0	0	0						
	6	0	0	0	0	0						
	7	0	0	0	0	0						
	8	0	0	0	0							

【図 41】

ディザマトリックスサイズ		画素群サイズ (画素)		ブロック 数	変換 テーブル サイズ (バイト)	状態数 ／ 画素群	使用 ビット 数	データ 圧縮率
主走査 方向	副走査 方向	主走査 方向	副走査 方向					
64	64	2	2	1024	256K	35	6	0.75
		4	2	512	128K	165	8	0.5
		4	4	256	128K	969	10	0.31
128	64	2	2	2048	512K	35	6	0.75
		4	2	1024	256K	165	8	0.5
		4	4	512	256K	969	10	0.31
128	128	2	2	4096	1024K	35	6	0.75
		4	2	2048	512K	165	8	0.5
		4	4	1024	512K	969	10	0.31

【図 4 2】

画素群サイズ (画素)		順序値 の個数	コード化さ れた個数 データの 個数	変換テーブ ルに必要な メモリ量 (Kバイト)
主走査 方向	副走査 方向			
2	2	4	35	0.034
4	2	8	165	0.322
4	4	16	969	3.785

## 【書類名】 要約書

## 【要約】

【課題】 高画質な画像を簡素な処理で迅速に出力可能な技術を提供する。

【解決手段】 画像を複数の画素群に分割し、画素群毎に画素群階調値を決定する。そして、画素群毎に付与されている分類番号と画素群階調値との組合せから、第1の変換テーブルを参照することで画素群の個数データを決定する。次いで、画素群内の各画素について順序値を記憶しておき、順序値と個数データとから第2の変換テーブルを参照することで、画素毎にドット形成の有無を決定してドットを形成する。個数データはデータ量が小さく迅速にデータをやり取り可能であり、第1の変換テーブルを参照すれば個数データを迅速に決定可能であり、第2の変換テーブルを参照すれば、画素毎のドット形成有無を迅速に決定可能である。このため高画質な画像データを迅速に出力することが可能となる。

【選択図】 図1

認定・付加情報

特許出願の番号	特願 2 0 0 4 - 0 8 2 7 0 9
受付番号	5 0 4 0 0 4 6 8 6 0 7
書類名	特許願
担当官	第三担当上席 0 0 9 2
作成日	平成 1 6 年 3 月 2 3 日

< 認定情報・付加情報 >

【提出日】 平成 1 6 年 3 月 2 2 日

特願 2 0 0 4 - 0 8 2 7 0 9

出 願 人 履 歴 情 報

識別番号 [ 0 0 0 0 0 2 3 6 9 ]

1. 変更年月日	1 9 9 0 年 8 月 2 0 日
[変更理由]	新規登録
住 所	東京都新宿区西新宿 2 丁目 4 番 1 号
氏 名	セイコーエプソン株式会社